

# Naruto Ultimate Ninja Storm

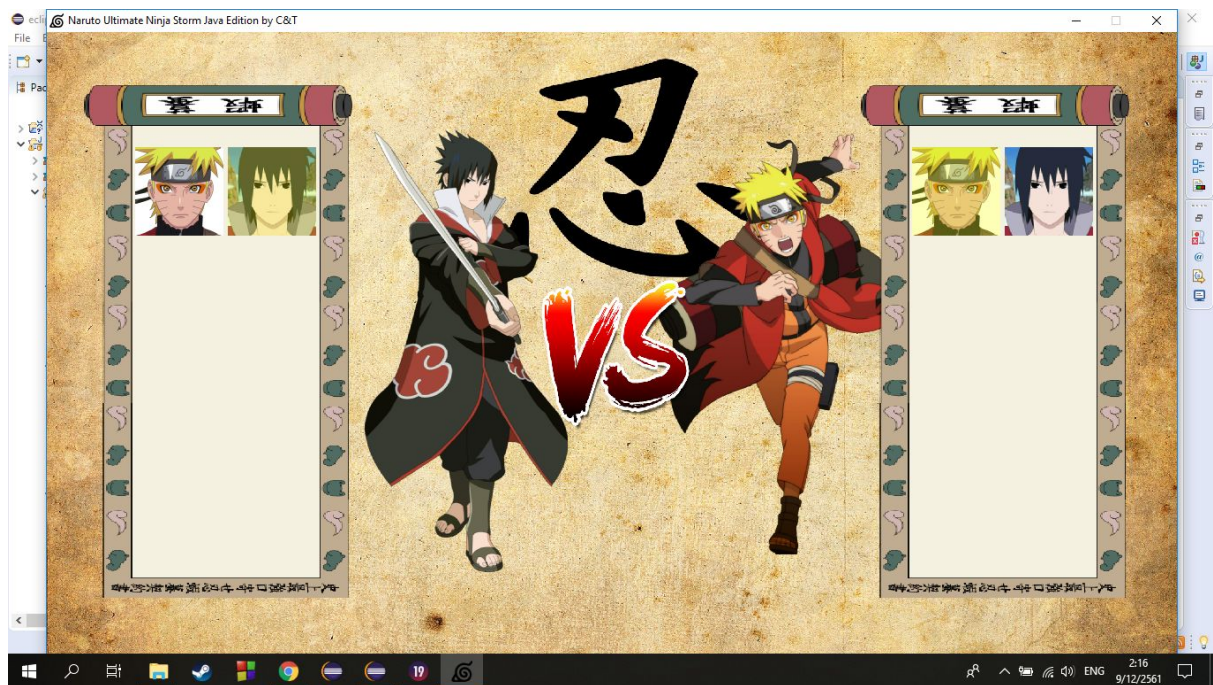
By

6030263321 Tana Chanchiew

6030272021 Thammakorn Kobkuachaiyapong

## 1. Overview

Naruto Ultimate Ninja Storm เป็นเกมต่อสู้ 2 มิติ ที่เลียนแบบมาจาก Naruto Mugen โดยนำโมเดลตัวละครจากการ์ตูนเรื่อง Naruto Shippuden(นารูโตะ ตำนานวายุสลาตัน) โดยผู้เล่นสามารถควบคุมตัวละครเพื่อใช้ต่อสู้กันเองในพื้นที่จำกัดได้



รูปที่ 1 Screenshot จากเกม

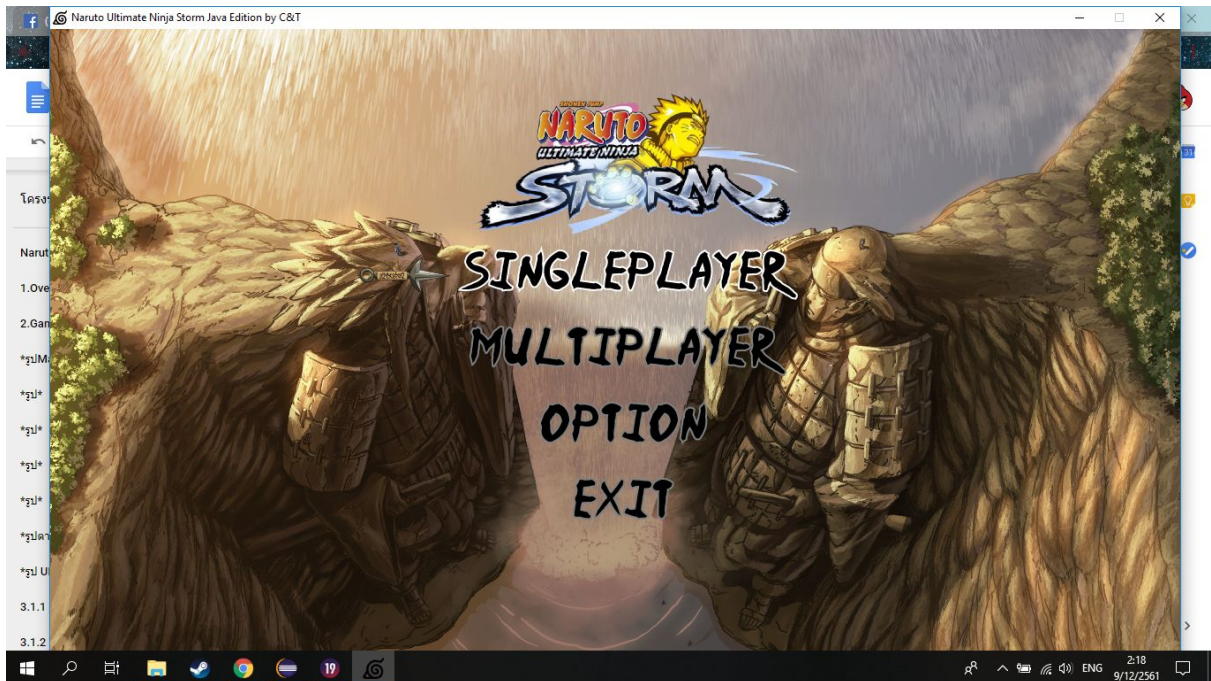
## 2.Gameplay

เมื่อเปิดเกม ตัวเกมจะทำการเปิดคลิป Intro และเล่นจนจบเข้าสู่หน้า Intro



รูปที่ 2 Screenshot จากหน้า Intro

จากนั้นเมื่อผู้เล่นกดปุ่มใดๆ ตัวเกมจะเข้าสู่หน้า Main Menu



รูปที่ 3 Screenshot จากหน้า Main Menu



โดยแต่ละหัวข้อของMainmenu นั้นจะเข้าสู่หน้าที่แตกต่างกันไป ดังนี้

- Singleplayer - (Coming Soon)เข้าสู่โหมดเล่นคนเดียวโดยสู้กับคอมพิวเตอร์ หรือออนไลน์
- Multiplayer - เข้าสู่หน้าเลือกตัวละคร เพื่อต่อสู้กับผู้เล่น(Offline)
- Option - เข้าสู่หน้าปรับคีย์ที่ใช้ในการเล่น
- Exit - ปิดเกม

เมื่อผู้เล่นเลือก Option จะเข้าสู่หน้าปรับคีย์ ซึ่งผู้เล่นสามารถปรับเปลี่ยนคีย์ที่ใช้ได้ตามที่ถนัด และจะรีเซ็ตทุกครั้งที่เปิดเกม โดยลำดับการตั้งปุ่มใหม่คือ เลือกคำสั่งที่ต้องการเปลี่ยน -> เลือกปุ่มที่ต้องการใช้



รูปที่ 4 Screenshot จากหน้า Option

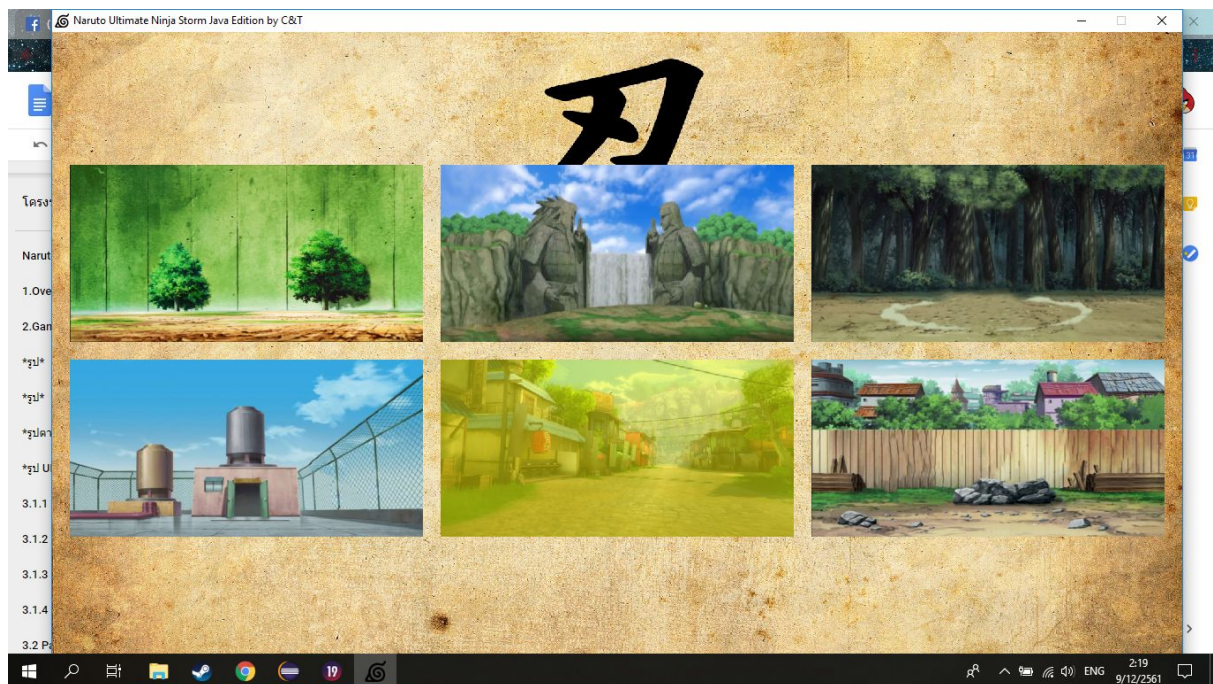
เมื่อตั้งคีย์เรียบร้อยแล้วให้กดปุ่ม ESC หรือ BACK SPACE หรือ range เพื่อย้อนกลับสู่หน้า Mainmenu

เมื่อผู้เล่นเลือก Multiplayer เกมจะเข้าสู่หน้าเลือกตัวละคร โดยในขณะนี้มิให้เลือก 2 ตัว



รูปที่ 5 Screenshot จากหน้า Multiplayer

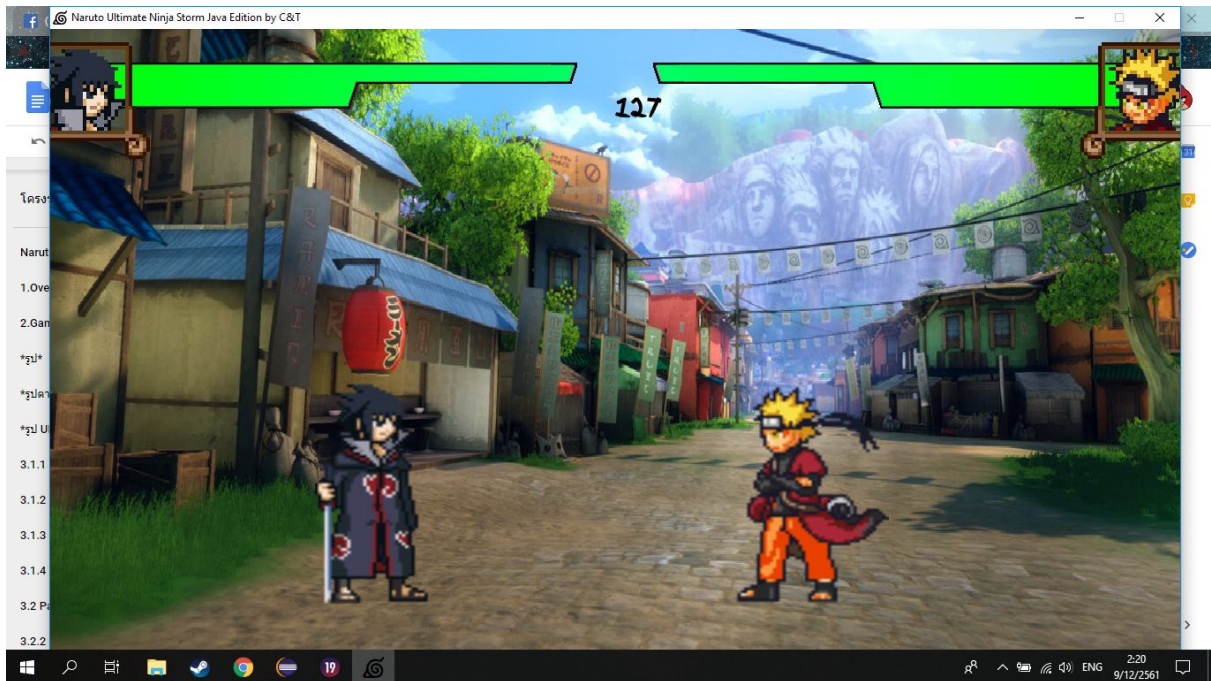
หลังจากที่ผู้เล่นทั้ง 2 คนเลือกตัวละครเสร็จสิ้นตัวเกมจะเข้าสู่หน้าเลือกแผนที่ โดยจะมีแผนที่ให้เลือกต่าง ๆ กันไปทั้งหมด 6 แบบ



รูปที่ 6 Screenshot จากหน้า Map Choose

เมื่อทำการเลือกตัวละครและแผนที่เสร็จสิ้นเกมจะเข้าสู่หน้าต่างต่อสู้

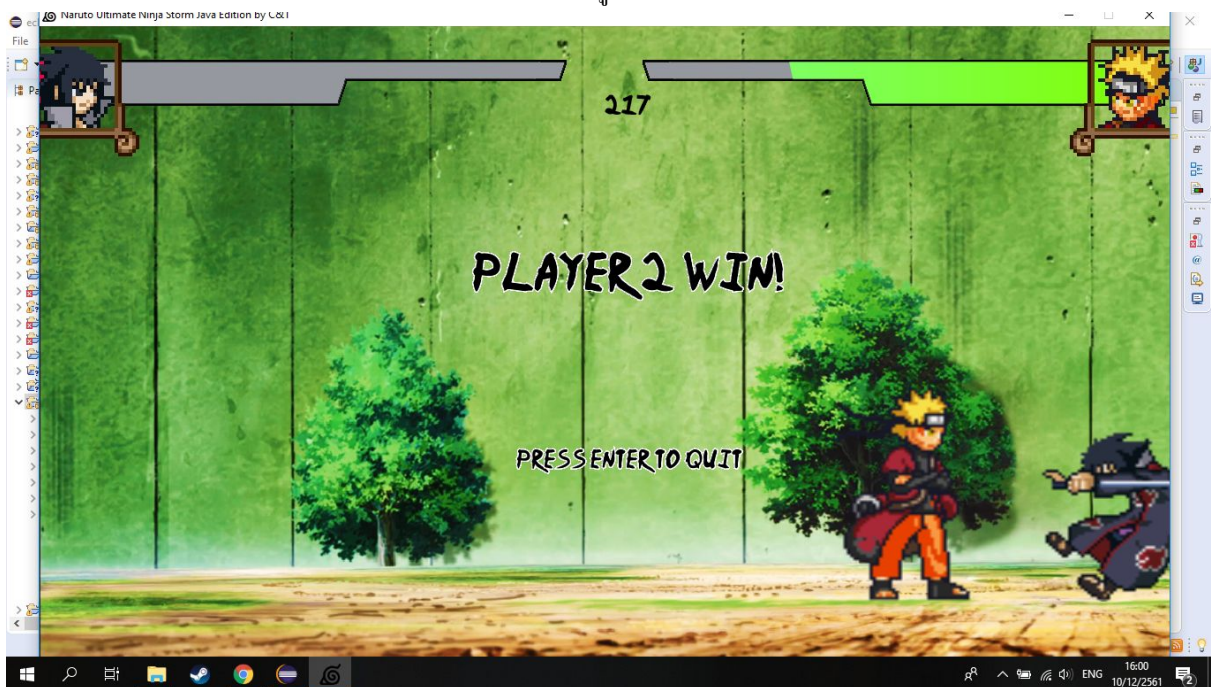




รูปที่ 7 Screenshot จากหน้า Game screen

ในหน้าต่างจะมีตัวจับเวลาและหลอดเลือด ซึ่งใช้ในการตัดสินเกมแบ่งเป็น 2 แบบคือ

1. เวลาหมด - ผู้ชนะคือผู้ที่เหลือเลือดมากกว่า
2. เลือดฝ่ายใดฝ่ายหนึ่งหมด - ฝ่ายที่ยังเหลือเลือดอยู่เป็นฝ่ายชนะ

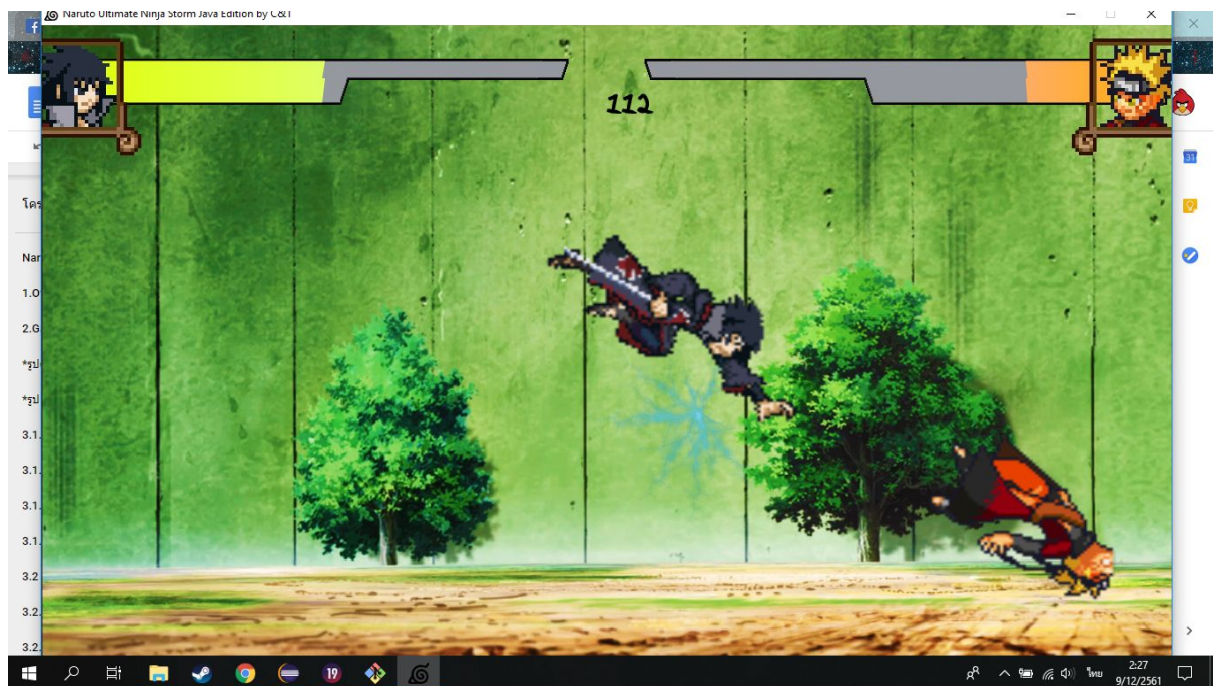


โดยผู้เล่นมีปุ่มควบคุมทั้งหมด 8 ปุ่ม ดังนี้

1. กดขึ้น (Default - W,Up) : เพื่อทำการกระโดด
2. กดลง (Default - S,Down) : เพื่อทำการย่อตัว
3. กดซ้าย (Default - A,Left) : เดิน/หันหน้าไปทางซ้าย
4. กดขวา (Default - D,Right) : เดิน/หันหน้าไปทางขวา
5. โจมตีระยะใกล้ (Default - J,Num 1)
6. โจมตีระยะไกล (Default - K,Num 2)
7. ป้องกัน (Default - L,Num 3)
8. หลบหลีก (Default - I,Num 5)

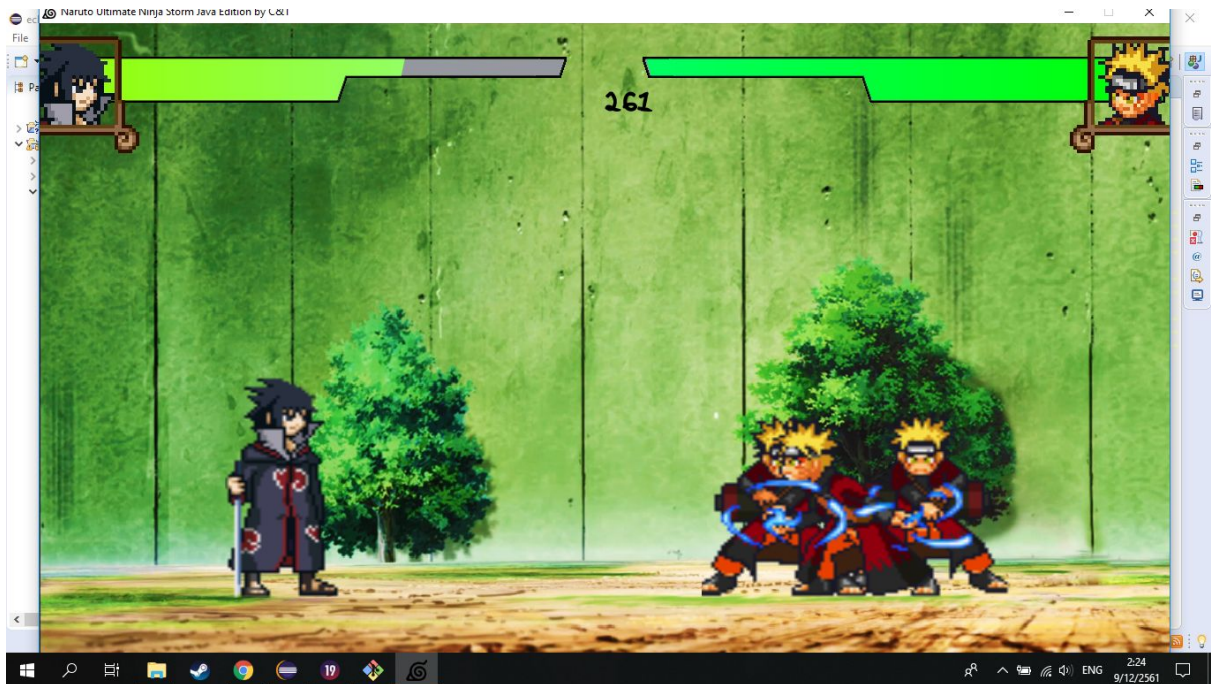
นอกจากนี้ ผู้เล่นสามารถกดปุ่มต่อเนื่องกันเพื่อ เรียกกระบวนท่าออกมา โดยแบ่งเป็น 3 ระดับ โดยผู้เล่นต้องฝึกฝนและหา Combination ที่ถูกต้องเอง

1. Basic skills
2. Mid skills
3. Advance skills



รูปที่ 8 Screenshot จากหน้า Game screen ตัวอย่างการใช้ Basic skills ของ Sasuke



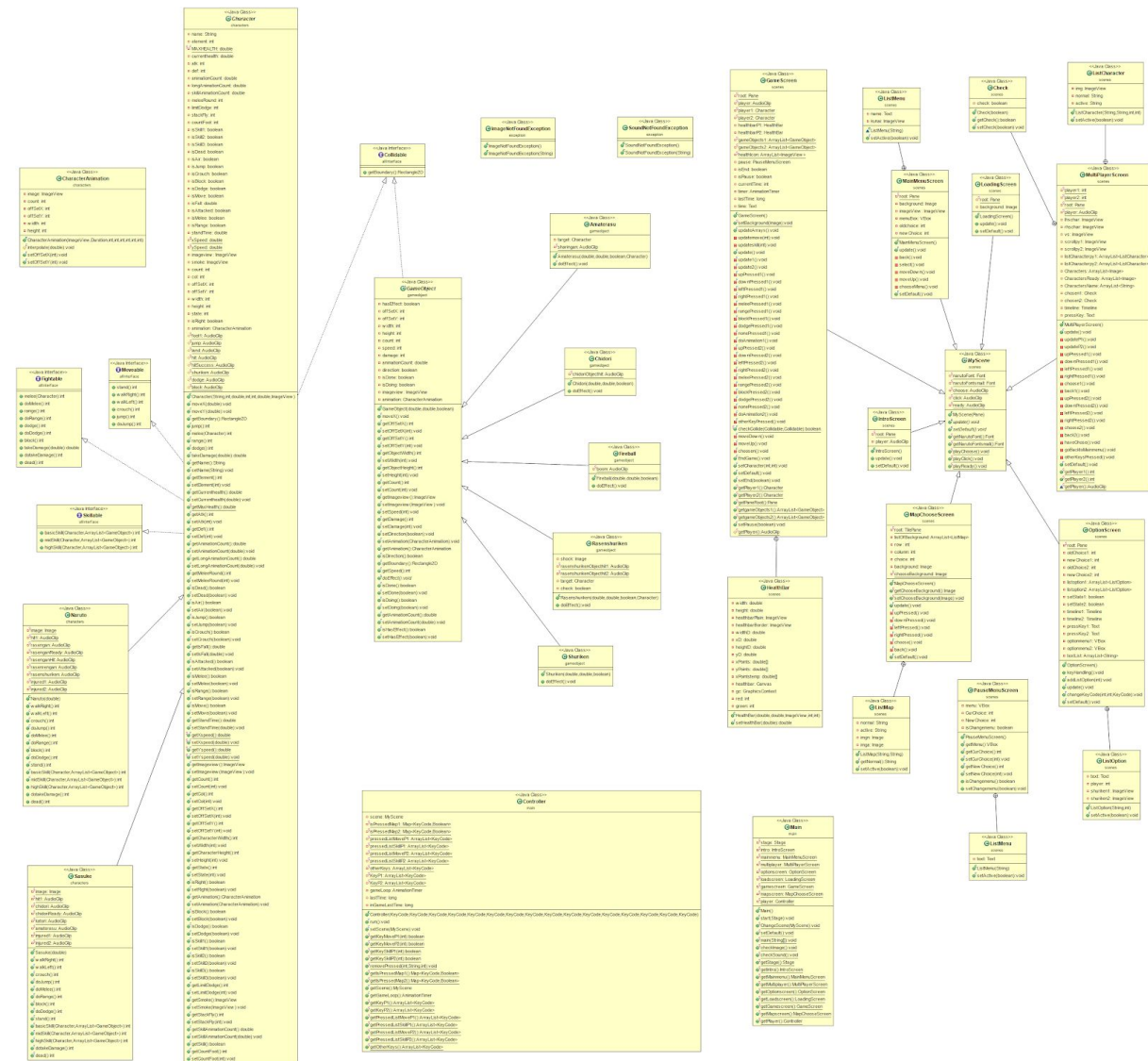


รูปที่ 9 Screenshot จากหน้า Game screen ตัวอย่างการใช้ Mid skills ของ Naruto



รูปที่ 10 Screenshot จากหน้า Game screen ตัวอย่างการใช้ Advance skills ของ Naruto

### 3.Implementation Details



### รูปที่ 10 UML Diagram ของโปรแกรม

### 3.1 Package allInterface - เก็บ interface ทั้งหมด

### 3.1.1 Interface Collidable

+Rectangle2D getBoundary()	
----------------------------	--

### 3.1.2 Interface Fightale

+int melee(Character target)	
+int doMelee()	
+int range()	
+int doRange()	



+int dodge()	
+int doDodge()	
+int block()	
+double takeDamage(double dmg)	
+int dotakeDamage()	
+int dead()	

### 3.1.3 Interface Skillable

+int basic_skill(Character target)	
+int mid_skill(Character target)	
+int High_skill(Character target)	

### 3.1.4 Interface Moveable

+int stand()	
+int walk_right()	
+int walk_left()	
+int couch()	
+int jump()	
+int doJump()	

## 3.2 *Package characters*

### 3.2.1 Abstract Class Character extends Pane implements Fightable, Moveable, Skillable, Collidable

-String name	เก็บชื่อตัวละคร
-int element	เก็บธาตุของตัวละคร
<u>-final double MAXHEALTH</u>	เก็บเลือดสูงสุดของตัวละคร
-double currenthealth	เก็บเลือดของตัวละคร
-int atk	เก็บพลังโจมตีของตัวละคร

-int def	เก็บพลังป้องกันของตัวละคร
-int animationCount	เก็บตัวเลขที่ใช้กำหนดช่วงภาพเคลื่อนไหว
-int longAnimationCount	เก็บตัวเลขที่ใช้กำหนดช่วงภาพเคลื่อนไหวที่มีขนาดยาว
-int skillAnimationCount	เก็บตัวเลขที่ใช้กำหนดช่วงภาพเคลื่อนไหวกระบวนท่า
-int melee_round	เก็บตัวเลขที่ใช้ในรอบของการโจมตีใกล้
-int limitDodge	เก็บขีดจำกัดครั้งของการหลบ
-int stackFly	เก็บตัวเลขที่ใช้ในรอบการถูกโจมตี
-int countFoot	เก็บตัวเลขที่ใช้ในรอบการก้าว
-boolean isSkill1	เก็บค่าสถานะการใช้ของกระบวนท่าที่ 1
-boolean isSkill2	เก็บค่าสถานะการใช้ของกระบวนท่าที่ 2
-boolean isSkill3	เก็บค่าสถานะการใช้ของกระบวนท่าที่ 3
-boolean isDead	เก็บค่าสถานะชีวิตของตัวละคร
-boolean isAir	เก็บค่าสถานะการอยู่บนอากาศของตัวละคร
-boolean isJump	เก็บค่าสถานะการกระโดดของตัวละคร
-boolean isCrouch	เก็บค่าสถานะการย่อตัวของตัวละคร
-boolean isBlock	เก็บค่าสถานะการป้องกันของตัวละคร
-boolean isDodge	เก็บค่าสถานะการหลบของตัวละคร
-boolean isMove	เก็บค่าสถานะการเคลื่อนที่ของตัวละคร
-double isFall	เก็บเวลาการล้มของตัวละคร
-boolean isAttacked	เก็บค่าสถานะการถูกโจมตีของตัวละคร
-boolean isMelee	เก็บค่าสถานะการโจมตีใกล้ของตัวละคร
-boolean isRange	เก็บค่าสถานะการโจมตีไกลของตัวละคร
-double standTime	เก็บเวลาการลุกของตัวละคร
-static double xSpeed	เก็บความเร็วมาตรฐานในแกน X ของตัวละคร
-static double ySpeed	เก็บความเร็วมาตรฐานในแกน Y ของตัวละคร
-ImageView imageview	เก็บ ImageView ของตัวละคร
-ImageView smoke	เก็บ ImageView ของรูปควัน
-int count	เก็บจำนวนเฟรมของอนิเมชันของตัวละคร



-int col	เก็บจำนวนคอลัมน์ของอนิเมชันของตัวละคร
-int offSetX	เก็บพิกัดจุดเริ่มต้นในแนวนอน X ของภาพอนิเมชันของตัวละคร
-int offSetY	เก็บพิกัดจุดเริ่มต้นในแนวแกน Y ของภาพอนิเมชันของตัวละคร
-int width	เก็บความกว้างของเฟรมตัวละคร
-int height	เก็บความสูงของเฟรมตัวละคร
-int state	เก็บสถานะของตัวละคร
-boolean isRight	เก็บสถานะการหันทางขวาของตัวละคร
-CharacterAnimation animation	เก็บคลาส CharacterAnimation เพื่อสร้างอนิเมชัน
#static AudioClip foot1	เก็บเสียงเท้า
#static AudioClip jump	เก็บเสียงกระโดด
#static AudioClip land	เก็บเสียงเมื่อลงพื้น
#static AudioClip hit	เก็บเสียงเมื่อโจมตี
#static AudioClip hitSuccess	เก็บเสียงเมื่อโจมตีสำเร็จ
#static AudioClip shuriken	เก็บเสียงเมื่อปาดาวกระจาย
#static AudioClip dodge	เก็บเสียงเมื่อหลบ
#static AudioClip block	เก็บเสียงเมื่อป้องกัน
+Character(String name, int element, double currentHealth, int atk, int def, double standTme,ImageView imageview)	Initializes name, element, currentHealth, atk, def, standTme and imageview.
+final void moveX(double d)	เคลื่อนตัวละครตามแกน X จำนวน d ครั้ง
+final void moveY(double d)	เคลื่อนตัวละครตามแกน Y จำนวน d ครั้ง
+final Rectangle2D getBoundary()	get Hitbox of characters
+final int jump()	Check if character can jump and set variables.
+final int melee(Character target)	Check if character can do melee attack and make character melee.
+final int range()	Check if character can do range attack and make character range.
+final int dodge()	Check if character can dodge and make

	character dodge.
+final double takeDamage(double dmg)	Check if character can dodge and set make character take damage.
Getters & Setters	Every fields except AudioClip variable type has its getter and setter. But MAXHEALTH has only getter.

### 3.2.2 Class CharacterAnimation extends Transition

-ImageView image	เก็บ ImageView ของตัวละคร
-int offSetX	เก็บพิกัดจุดเริ่มต้นในแนวแกน X ของภาพอนิเมชันของตัวละคร
-int offSetY	เก็บพิกัดจุดเริ่มต้นในแนวแกน Y ของภาพอนิเมชันของตัวละคร
-int width	เก็บความกว้างของเฟรมตัวละคร
-int height	เก็บความสูงของเฟรมตัวละคร
+CharacterAnimation(ImageView image, Duration duration, int count, int col, int offSetX, int offSetY,int width, int height)	Initializes every field of the class
#final void interpolate(double frac)	This method has to be provided by implementations of Transition. While a Transition is running, this method is called in every frame. The parameter defines the current position with the animation. At the start, the fraction will be 0.0 and at the end it will be 1.0.
Setters	Setter of offSetX and offSetY.

### 3.2.3 Class Naruto extends Character

-static Image image	เก็บรูปของ Naruto
-static AudioClip hit1	เก็บเสียงพูดของ Naruto เมื่อโจมตี
-static AudioClip rasengan	เก็บเสียงพูดของ Naruto เมื่อสร้าง Rasengan
-static AudioClip rasenganReady	เก็บเสียงของ Rasengan
-static AudioClip rasenganHit	เก็บเสียงเมื่อ Rasengan ชนผู้เล่นฝั่งตรงข้าม
-static AudioClip rasenrenzan	เก็บเสียงพูดของ Naruto เมื่อสร้าง Rasenrenzan



-static AudioClip rasenshuriken	เก็บเสียงพูดของ Naruto เมื่อสร้าง Rasenshuriken
-static AudioClip injured1	เก็บเสียงพูดของ Naruto เมื่อถูกโจมตี แบบที่ 1
-static AudioClip injured2	เก็บเสียงพูดของ Naruto เมื่อถูกโจมตี แบบที่ 2
+Naruto(double currentHealth)	Initializes every field of the class by superclass
+final int walkRight()	Check if Naruto can do melee attack and make him walk in right direction.
+final int walkLeft()	Check if Naruto can do melee attack and make him walk in left direction.
+final int crouch()	Check if Naruto can do melee attack and make him crouch.
+final int doJump()	Show Naruto's jump animation.
+final int doMelee()	Show Naruto's melee attack animation.
+final int doRange()	Show Naruto's range attack animation.
+final int block()	Check if Naruto can block and make character crouch.
+final int doDodge()	Show Naruto's dodge animation.
+final int stand()	Check if Naruto can stand and make him stand.
+final int basicSkill(Character target, ArrayList<GameObject> playerObject)	Check if Naruto can do skill 1 and make him do skill 1.
+final int midSkill(Character target, ArrayList<GameObject> playerObject)	Check if Naruto can do skill 2 attack and make him do skill 2.
+int highSkill(Character target, ArrayList<GameObject> playerObject)	Check if Naruto can do skill 3 attack and make him do skill 3.
+int dotakeDamage()	Show Naruto's take damage animation.
+int dead()	Check if Naruto dead and make him dead.

### 3.2.4 Class Sasuke extends Character

-static Image image	เก็บรูปของ Sasuke
---------------------	-------------------

-static AudioClip hit1	เก็บเสียงพูดของ Sasuke เมื่อโจมตี
-static AudioClip chidori	เก็บเสียงพูดของ Sasuke เมื่อสร้าง Chidori
-static AudioClip chidoriReady	เก็บเสียงของ Chidori
-static AudioClip katon	เก็บเสียงพูดของ Sasuke เมื่อสร้าง Katon
-static AudioClip amaterasu	เก็บเสียงพูดของ Sasuke เมื่อสร้าง Amaterasu
-static AudioClip injured1	เก็บเสียงพูดของ Sasuke เมื่อถูกโจมตี แบบที่ 1
-static AudioClip injured2	เก็บเสียงพูดของ Sasuke เมื่อถูกโจมตี แบบที่ 2
+Sasuke(double currentHealth)	Initializes every field of the class by superclass
+final int walkRight()	Check if Sasuke can do melee attack and make him walk in right direction.
+final int walkLeft()	Check if Sasuke can do melee attack and make him walk in left direction.
+final int crouch()	Check if Sasuke can do melee attack and make him crouch.
+final int doJump()	Show Sasuke's jump animation.
+final int doMelee()	Show Sasuke's melee attack animation.
+final int doRange()	Show Sasuke's range attack animation.
+final int block()	Check if Sasuke can block and make him crouch.
+final int doDodge()	Show Sasuke's dodge animation.
+final int stand()	Check if Sasuke can stand and make him stand.
+final int basicSkill(Character target, ArrayList<GameObject> playerObject)	Check if Sasuke can do skill 1 and make him do skill 1.
+final int midSkill(Character target, ArrayList<GameObject> playerObject)	Check if Sasuke can do skill 2 and make him do skill 2.
+int highSkill(Character target, ArrayList<GameObject> playerObject)	Check if Sasuke can do skill 3 and make him do skill 3.
+int dotakeDamage()	Show Sasuke's take damage animation.
+int dead()	Check if Sasuke dead and make him dead.

### 3.3 Package gameObject



### 3.3.1 Abstract Class GameObject extends Pane implements Collidable

-boolean hasEffect	เก็บค่าสถานะว่าวัตถุมีเอฟเฟกต์หรือไม่
-int offSetX	เก็บพิกัดจุดเริ่มต้นในแนวแกน X ของภาพอนิเมชันของวัตถุ
-int offSetY	เก็บพิกัดจุดเริ่มต้นในแนวแกน Y ของภาพอนิเมชันของวัตถุ
-int width	เก็บความกว้างของวัตถุ
-int height	เก็บความสูงของวัตถุ
-int count	เก็บจำนวนเฟรมของอนิเมชันของวัตถุ
-int speed	เก็บความเร็วของอนิเมชันของวัตถุ
-int damage	เก็บความเสียหายที่เกิดจากวัตถุ
-double animationCount	เก็บตัวเลขที่ใช้กำหนดช่วงภาพเคลื่อนไหว
-boolean direction	เก็บทิศทางของวัตถุ
-boolean isDone	เก็บสถานะของวัตถุหากทำงานเสร็จสิ้น
-boolean isDoing	เก็บสถานะของวัตถุหากวัตถุกำลังทำงาน
-ImageView imageview	Imagview ของวัตถุ
-CharacterAnimation animation	คลาส CharacterAnimation ของวัตถุ
+GameObject(double posx, double posy, boolean direction)	Initializes every field of the class by superclass
+final void moveX()	Make object move in X-axis
+final Rectangle2D getBoundary()	
+abstract void doEffect()	If object has animation, show it.
Getters & Setters	Every fields has its getter and setter.

### 3.3.2 Class Amaterasu extends GameObject

-Character target	เก็บ Character ที่เป็นเป้าหมายของ Amaterasu
+Chidori(double posx, double posy, boolean direction)	Initializes every field of the class by superclass
+void doEffect()	Show Amaterasu's animation.

### 3.3.3 Class Chidori extends GameObject

-static AudioClip chidoriObjecthit	เก็บเสียงเมื่อ Chidori ชนผู้เล่นฝั่งตรงข้าม
+Amaterasu(double posX, double posY, boolean direction, Character target)	Initializes every field of the class by superclass
+void doEffect()	Show Chidori's animation.

### 3.3.4 Class Fireball extends GameObject

-static AudioClip boom	เก็บเสียงเมื่อ Fireball ชนผู้เล่นฝั่งตรงข้าม
+Fireball(double posX, double posY, boolean direction)	Initializes every field of the class by superclass
+void doEffect()	Show Fireball's animation.

### 3.3.5 Class Rasenshuriken extends GameObject

-Image shock	เก็บภาพเมื่อ Rasenshuriken โดนเป้าหมาย
-static AudioClip rasenshurikenObjecthit1	เก็บเสียงเมื่อ Rasenshuriken ชนผู้เล่นฝั่งตรงข้าม แบบที่ 1
-static AudioClip rasenshurikenObjecthit2	เก็บเสียงเมื่อ Rasenshuriken ชนผู้เล่นฝั่งตรงข้าม แบบที่ 2
-Character target	เก็บ Character ที่เป็นเป้าหมายของ Rasenshuriken
-boolean check	เก็บค่าสถานะหาก Rasenshuriken ชนผู้เล่นฝั่งตรงข้ามแล้ว
+Rasenshuriken(double posX, double posY, boolean direction, Character target)	Initializes every field of the class by superclass
+void doEffect()	Show Rasenshuriken's animation.

### 3.3.6 Class Shuriken extends GameObject

+Shuriken(double posX, double posY, boolean direction)	Initializes every field of the class by superclass
+void doEffect()	This method do nothing. But we have to override it.

## 3.4 Package main

### 3.4.1 Class Controller implements Runnable

คลาสนี้ใช้ในการทำ KeyHandling เพื่อใช้ควบคุมส่วนต่างๆในเกมทั้งหมด

#### 3.4.1.1 fields

-MyScene scene	หน้าต่างเกมที่จะให้controllerทำงาน
-Map<KeyCode, Boolean> isPressedMap1	เป็นที่เก็บข้อมูลว่าปุ่มไหนกำลังถูกกดอยู่ สำหรับผู้เล่นคนที่ 1
-Map<KeyCode, Boolean> isPressedMap2	เป็นที่เก็บข้อมูลว่าปุ่มไหนกำลังถูกกดอยู่ สำหรับผู้เล่นคนที่ 2
-ArrayList<KeyCode> pressedListMoveP1	เก็บลำดับปุ่มเคลื่อนที่ของผู้เล่นที่ 1
-ArrayList<KeyCode> pressedListSkillP1	เก็บลำดับปุ่มกระบวนท่าของผู้เล่นที่ 1
-ArrayList<KeyCode> pressedListMoveP2	เก็บลำดับปุ่มเคลื่อนที่ของผู้เล่นที่ 2
-ArrayList<KeyCode> pressedListSkillP2	เก็บลำดับปุ่มกระบวนท่าของผู้เล่นที่ 2
-ArrayList<KeyCode> otherKeys	เก็บลำดับปุ่มอื่นๆ
-ArrayList<KeyCode> KeyP1	เก็บปุ่มบังคับผู้เล่นที่ 1
-ArrayList<KeyCode> KeyP2	เก็บปุ่มบังคับผู้เล่นที่ 2
-AnimationTimer gameloop	game's loop
-long lastTime	เวลาที่updateครั้งสุดท้ายในลูปเกมเพื่อติดตามผลลัพธ์
-long inGameLastTime	เวลาที่ update ครั้งสุดท้ายในลูปเกมเพื่อเช็คการทำคอมโบ

#### 3.4.1.2 Constructor

+Controller(KeyCode upKey_1, KeyCode downKey_1, KeyCode leftKey_1, KeyCode rightKey_1, KeyCode meleeKey_1, KeyCode rangeKey_1, KeyCode blockKey_1, KeyCode dodgeKey_1, KeyCode upKey_2, KeyCode downKey_2, KeyCode leftKey_2, KeyCode rightKey_2, KeyCode meleeKey_2, KeyCode rangeKey_2, KeyCode blockKey_2, KeyCode dodgeKey_2)	set default keys and initialize fields
---	--

#### 3.4.1.3 Methods



+final void run()	Run gameloop & Key's handling
+void removePressed(int player,String type, int amount)	ลบลำดับการกดปุ่มที่ได้ตรวจสอบไปแล้ว
Getters & Setters	Every fields has its getter. But only these fields are also have setter <ul style="list-style-type: none"> <li>• scene</li> </ul>

### 3.4.2 Class Main extends Application

ส่วนแสดงผลเกม

#### 3.4.2.1 Fields

-Stage stage	
-IntroScreen intro	
-MainMenuScreen mainmenu	
-MultiPlayerScreen multiplayer	
-OptionScreen optionscreen	
-LoadingScreen loadscreen	
-GameScreen gamescreen	
-MapChooseScreen mapscreen	
-Controller player	

#### 3.4.2.2 Methods

+void main(String[] args)	main method
+static void checkImage()	check if all image are available.
+static void checkSound()	check if all sound are available.
+final void start(Stage primaryStage)	start application
+void ChangeScene(MyScene nextScene)	ใช้ในการเปลี่ยน scene ของเกม
+void setDefault()	ตั้งค่าทั้งหมดกลับสู่สถานะเริ่มต้น
Getters	Every fields has its getter

## 3.5 Package Scenes

### 3.5.1 Abstract Class MyScene extends Scene

#### 3.5.1.1 Fields

<u>-Font narutoFont</u>	ฟอนต์ที่ใช้ในเกม
<u>-Font narutoFontsmall</u>	ฟอนต์ที่ใช้ในเกม
<u>-AudioClip choose</u>	เสียงการกดเลือก
<u>-AudioClip click</u>	เสียงเปลี่ยนเมนู
<u>-AudioClip ready</u>	เสียงเตรียมพร้อม

#### 3.5.1.2 Constructor

<u>+MyScene(Pane root)</u>	
----------------------------	--

#### 3.5.1.3 Methods

<u>+void update()</u>	
<u>+void setDefault()</u>	
<u>+Font getNarutoFont()</u>	
<u>+Font getNarutoFontsmall()</u>	
<u>+void playChoose()</u>	เล่นเสียง choose
<u>+void playClick()</u>	เล่นเสียง click
<u>+void playReady()</u>	เล่นเสียง ready

### 3.5.2 Class GameScreen extends MyScene

#### 3.5.2.1 Fields

<u>-Pane root</u>	GameScreen's pane
<u>-staticAudioClip player</u>	Background Music
<u>-static Character player1</u>	Player 1's character
<u>-static Character player2</u>	Player 2's character
<u>-HealthBar healthbaP1</u>	Player 1's health bar
<u>-HealthBar healthbaP2</u>	Player 2's health bar
<u>-ArrayList&lt;GameObject&gt; gameObjects1</u>	Player 1's Object
<u>-ArrayList&lt;GameObject&gt; gameObjects2</u>	Player 2's Object
<u>-ArrayList&lt;ImageView&gt; healthIcon</u>	Health bar's icon
<u>-boolean isEnd</u>	เก็บว่าเกมจบหรือไม่

-boolean isPause	กดหยุดเกมอยู่หรือไม่
-PauseMenuScreen pause	หน้าจอหยุดเกม
-int currentTime	ตัวจับเวลาในเกม
-AnimationTimer timer	ตัวจับเวลาในเกม
-long lastTime	ตัวอัปเดตเวลาในเกม
-Text time	ตัวอักษรบอกเวลา

### 3.5.2.2 Constructor

+GameScreen()	Initialize needed fields
---------------	--------------------------

### 3.5.2.3 Methods

<u>+final boolean checkCollide(Collidable obj1, Collidable obj2)</u>	เช็ค Object collision
<u>+void setBackground(Image background)</u>	set GameScreen background
+final void update()	update game
+final void updateArrays()	update combo checker
-final void update1()	update Player 1's keys
-final void update2()	update Player 2's keys
-final void upPressed1()	check if Player 1's pressed up
-final void downPressed1()	check if Player 1's pressed down
-final void leftPressed1()	check if Player 1's pressed left
-final void rightPressed1()	check if Player 1's pressed right
-final void meleePressed1()	check if Player 1's pressed melee
-final void rangePressed1()	check if Player 1's pressed range
-final void blockPressed1()	check if Player 1's pressed block
-final void dodgePressed1()	check if Player 1's pressed dodge
-final void nonePressed1()	check if Player 1's don't pressed any key
-final void upPressed2()	check if Player 2's pressed up
-final void downPressed2()	check if Player 2's pressed down
-final void leftPressed2()	check if Player 2's pressed left



-final void rightPressed2()	check if Player 2's pressed right
-final void meleePressed2()	check if Player 2's pressed melee
-final void rangePressed2()	check if Player 2's pressed range
-final void blockPressed2()	check if Player 2's pressed block
-final void dodgePressed2()	check if Player 2's pressed dodge
-final void nonePressed2()	check if Player 2's don't pressed any key
-void updateMove(int player)	check move's keys order
-void updateSkill(int player)	check skill's keys order
-final void doAnimation1()	Draw animation of Player 1
-final void doAnimation2()	Draw animation of Player 2
-final void otherKeyPressed()	check other keys
-final void moveDown()	move menu highlighter in pause
-final void moveUp()	move menu highlighter in pause
-final void choose	choose menu in pause screen
-final void EndGame()	This method call when the game is end
+final void setCharacter(int choose1, int choose2)	Generater character due to player chosen in Multiplayer screen
+final void setDefault()	set this screen to default
Getters & Setters	Getters & Setters of needed fields

### 3.5.3 SubClass GameScreen.HealthBar extends StackPane

#### 3.5.3.1 Fields

-double width	width of pane
-double height	height of pane
-ImageView healthbarPlain	Healthbar's Image
-ImageView healthbarBorder	Healthbar's Image
-double widthD	Healthbar's width
-double xD	Healthbar's x position
-double heightD	Healthbar's height
-double yD	Healthbar's y position

-double[] xPoints	Health's polygon x coordinate
-double[] yPoints	Health's polygon y coordinate
-double[] xPointstemp	Healthbar's polygon x coordinate
-Canvas healthbar	Polygon canvas
-GraphicsContext gc	Healthbar's graphic context
-int red	Red value
-int green	Green value

### 3.5.3.2 Constructor

+HealthBar(double width, double height, ImageView characters, int xpos, int ypos)	Initialize & draw healthbar
---	-----------------------------

### 3.5.3.3 Method

+final double setHealthBar(double curDIVmax)	adjust Healthbar curDIVmax is currenthealth/maxhealth
--	--

## 3.5.4 Class IntroScreen extends MyScene

### 3.5.4.1 Fields

- <u>Pane root</u>	Introscreen's pane
-AudioClip player	Introscreen's music

### 3.5.4.2 Constructor

+IntroScreen()	Initialize and make graphic
----------------	-----------------------------

### 3.5.4.3 Methods

+void update()	Press any key to skip intro
+void setDefault()	set Introscreen default

## 3.5.5 Class LoadingScreen extends MyScene

### 3.5.5.1 Fields

- <u>Pane root</u>	LoadingScreen's pane
-Image background	Loading gif

### 3.5.5.2 Constructor

+LoadingScreen()	set loading screen
------------------	--------------------

### 3.5.6 Class MainMenuScreen extends MyScene

#### 3.5.6.1 Fields

- <u>Pane root</u>	MainMenuScreen's pane
-Image background	Background image
-ImageView imageView	Logo
-VBox menuBox	Menu item container
-int oldchoice	active item index
-int newchoice	next active item index

#### 3.5.6.2 Constructor

+MainmenuScreen()	Initialize & generate graphics
-------------------	--------------------------------

#### 3.5.6.3

+final void update()	update screen
-void back()	Check if player pressed back, escape or range then return to intro screen
-void select()	Check if player pressed enter spacebar or melee
-void moveDown()	Check if player pressed down or left keys then move highlighter
-void moveUp()	Check if player pressed up or right keys then move highlighter
-void chooseMenu()	Check that what player choose then do what have assign
+final setDefault	set Screen to default

### 3.5.7 SubClass MainMenuScreen.ListMenu extends HBox



#### 3.5.7.1 Fields

-Text name	name of this item
-ImageView kunai	Image of pointer

#### 3.5.7.2 Constructor

ListMenu(String text)	Initialize & generate graphic
-----------------------	-------------------------------

#### 3.5.7.3 Method

+final void setActive(boolean check)	set pointer visible & change text's color
--------------------------------------	---

### 3.5.8 Class MapChooseScreen extends MyScene

#### 3.5.8.1 Fields

- <u>TilePane root</u>	Class's pane
-ArrayList<ListMap> listOfBackground	Listbackground's container
-int row	row
-int column	column
-int choice	index of Players selected

#### 3.5.8.2 Constructor

+MapChooseScreen()	Initialize & generat graphics
--------------------	-------------------------------

#### 3.5.8.3 Methods

+final void update()	update Screen
-final void upPressed()	Check if player pressed up keys then move highlighter
-final void downPressed()	Check if player pressed down keys then move highlighter
-final void leftPressed()	Check if player pressed left keys then move highlighter
-final void rightPressed()	Check if player pressed right keys then move highlighter
-final void choose()	Check that what player choose then do what have assign
-void back()	Check if player pressed back, escape or range then return to intro screen

+final setDefault	set Screen to default
Getters & Setters	Getters & Setters of needed fields

### 3.5.9 SubClass MapChooseScreen.ListMap extends ImageView

#### 3.5.9.1 Fields

-String normal	normal stage's image directory
-String active	active stage's Image directory
-Image imgn	normal stage's image
-Image imga	active stage's Image

#### 3.5.9.2 Constructor

+ListMap(String n ,String a)	Initialize & load images
------------------------------	--------------------------

#### 3.5.9.3 Methods

+final void setActive(coolean check)	activate or deactivate image
+final String getNormal()	normal's getter

### 3.5.10 OptionScreen extends MyScene

#### 3.5.10.1 Fields

- <u>Pane root</u>	Class's pane
-int oldChoice1	Player 1's activating item
-int newChoice1	Player 1's next active item
-int oldChoice2	Player 2's activating item
-int newChoice2	Player 2's next active item
-ArrayList<ListOption> listoption1	Player 1's Listoption container
-ArrayList<ListOption> listoption2	Player 2's Listoption container
-boolean setState1	Player 1's state of setting key
-boolean setState2	Player 2's state of setting key
-Timeline timeline1	timeline of text
-Timeline timeline2	timeline of text
-Text pressKey1	text to tell what player can do

-Text pressKey2	text to tell what player can do
-VBox optiomenu1	Player 1's menu item container
-VBox optiomenu2	Player 2's menu item container
-ArrayList<String> textList	Key's text list

#### 3.5.10.2 Constructor

+OptionScreen()	Initialize & generate graphics
-----------------	--------------------------------

#### 3.5.10.3 Methods

+void keyHandling()	handle setting keys
+final void addListOption(int player)	initialize list of option
+final void update()	update screen
+final void changeKeyCode(int player, int choice, KeyCode key)	method for set key
+final void setDefault	set fields to default

### 3.5.11 SubClass ListOption extends HBox

#### 3.5.11.1 Fields

-Text text	Text of item
-int player	player integer
-ImageView shuriken1	Shuriken's image
-ImageView shuriken2	Shuriken's image

#### 3.5.11.2 Constructor

+ListOption(String text,int player)	Initialize & generate graphics
-------------------------------------	--------------------------------

#### 3.5.11.3 Method

+final void setActive(boolean check)	activate or deactivate item
--------------------------------------	-----------------------------

### 3.5.12 MultiplayerScreen extends MyScene

#### 3.5.12.1 Fields

<u>-int player1</u>	Player 1's select character
<u>-int player2</u>	Player 2's select character
<u>-Pane root</u>	Class pane
<u>-static AudioClip player</u>	Background music



-ImageView lhschar	Player 1's image of character
-ImageView rhschar	Player 2's image of character
-ImageView vs	VS's image
-ImageView scrollpy1	Scroll's image
-ImageView scrollpy2	Scroll's image
-ArrayList<ListCharacter> listCharacterpy1	Player 1's list character container
-ArrayList<ListCharacter> listCharacterpy2	Player 2's list character container
-ArrayList<Image> Characters	Character's image container
-ArrayList<Image> CharactersReady	Character ready's image container
-ArrayList<String> CharactersName	Character's name container
-Check chosen1	Check that player 1 have chosen or not
-Check chosen2	Check that player 2 have chosen or not
-Timelinr timeline	Text's timeline
-Text pressKey	Text

### 3.5.12.2 Constructor

+MultiPlayerScreen()	Initialize & generate graphics
----------------------	--------------------------------

### 3.5.12.3 Methods

+final void update()	update game
-final void updateP1()	update Player 1's keys
-final void updateP2()	update Player 2's keys
-final void upPressed1()	check if Player 1's pressed up
-final void downPressed1()	check if Player 1's pressed down
-final void leftPressed1()	check if Player 1's pressed left
-final void rightPressed1()	check if Player 1's pressed right
-final void choose1()	Check what character Player 1 choose and change left side character image
-final void back1()	Undo Player 1's choose or back to mainmenu
-final void upPressed2()	check if Player 2's pressed up

-final void downPressed2()	check if Player 2's pressed down
-final void leftPressed2()	check if Player 2's pressed left
-final void rightPressed2()	check if Player 2's pressed right
-final void choose2()	Check what character Player 2 choose and change left side character image
-final void back2()	Undo Player 2's choose or back to mainmenu
-void haveChose()	Check if both player have choose then change scene
-void goBacktoMainmrnu()	back to main menu
-final void otherKeyPressed()	check other keys
+final void setDefault()	set this screen's field to default
Getters	Getters of needed fields

### 3.5.13 SubClass MultiPlayerScreen.ListCharacter extends HBox

#### 3.5.13.1 Fields

-ImageView img	img of character
-String normal	normal stage's image directory
-String active	active stage's Image directory

#### 3.5.13.2 Constructor

ListCharacter(String n, String a, int posx, int posy)	Initialize & generate graphics
---	--------------------------------

#### 3.5.13.3 Method

+final void setActive(boolean check)	activate or deactivate items
--------------------------------------	------------------------------

### 3.5.14 SubClass MultiPlayerScreen.Check

#### 3.5.14.1 Fields

-boolean check	checker
----------------	---------

#### 3.5.14.2 Constructor

+Check(boolean c)	assign check
-------------------	--------------

### 3.5.14.3 Methods

Getter & Setter	Getter & Setter of check
-----------------	--------------------------

## 3.5.15 PauseMenuScreen extends Pane

### 3.5.15.1 Fields

-VBox menu	menu item's container
-int CurChoice	activating item index
-int NewChoice	next activate item index
-boolean isChangemenu	continuous pressed key handler

### 3.5.15.2 Constructor

+PauseMenuScreen()	Initialize & generate graphics
--------------------	--------------------------------

### 3.5.15.3 Methods

Getters & Setters	Getters & Setters of needed fields
-------------------	------------------------------------

## 3.5.16 SubClass Listmenu extend HBox

### 3.5.16.1 Fields

-Text text	text of items
------------	---------------

### 3.5.16.2 Constructor

+ListMenu(String text)	Initialize & generate graphics
------------------------	--------------------------------

### 3.5.16.3 Method

+final void setActive(boolean check)	activate or deactivate item
--------------------------------------	-----------------------------

### 3.6 *Package exception* - เก็บ exception ทั้งหมด

#### 3.1.1 Class ImageNotFoundException

+SoundNotFoundException()	Initialize exception
+SoundNotFoundException(String message)	Initialize exception with string

#### 3.1.2 Class SoundNotFoundException

+ImageNotFoundException()	Initialize exception
+ImageNotFoundException(String message)	Initialize exception with string