

5 Knights At Thonny Documentation

จัดทำโดย

นาย ภูวนัช เพิ่มผล 6430315221

นาย วรเมธ สุวรรณพฤษ 6431340021

2110215 Programming Methodology

Semester 1 Year 2022

Chulalongkorn University

5 Knights at Thonny

ความเป็นมา

5 knights at thonny มีแรงบันดาลใจมาจากเกม 7 knights ซึ่งเป็นเกม turn-base สลับกันโจมตีกับบอทที่เล่นเป็นฝ่ายตรงข้าม ซึ่งจุดมุ่งหมายของเกมนี้คือ อยาให้ผู้เล่นวางแผนการเลือกตัวละครเพื่อไปโจมตีฝ่ายตรงข้ามที่มีเลเวลมากขึ้นเรื่อย ๆ

กฎกติกา

ในตอนเริ่มเกมผู้เล่นจะเริ่มจากการเลือกตัวละครลงไปในสนาม โดยผู้เล่นมีตัวละคร 5 ชนิด ให้เลือกลงสนามได้ 3 ตัว และหลังจากเลือกเสร็จก็จะเข้าเทิร์นของผู้เล่น โดยผู้เล่นจะสามารถเลือกใช้งานความสามารถของตัวละคร หรือ ความสามารถของ potion ได้ก่อนแล้วค่อยเลือกโจมตีฝ่ายตรงข้ามและเมื่อผู้เล่นโจมตีเสร็จแล้วฝ่ายตรงข้ามจะสุมตัวเพื่อโจมตีเรากลับ แล้วก็จะสลับเทิร์นโจมตีกันไปเรื่อย ๆ จนฝ่ายใดฝ่ายหนึ่งตายหมด ฝ่ายที่เหลือตัวละครรอดอยู่เป็นฝ่ายสุดท้ายจะถือว่าเป็นผู้ชนะ และเมื่อผู้เล่นชนะจะได้รับเงินและสามารถเข้าไปซื้อไอเทมหรือเลือกอัปเลเวลตัวละครได้

อธิบายไอเทมและตัวละคร

ตัวละคร มี 5 ชนิด

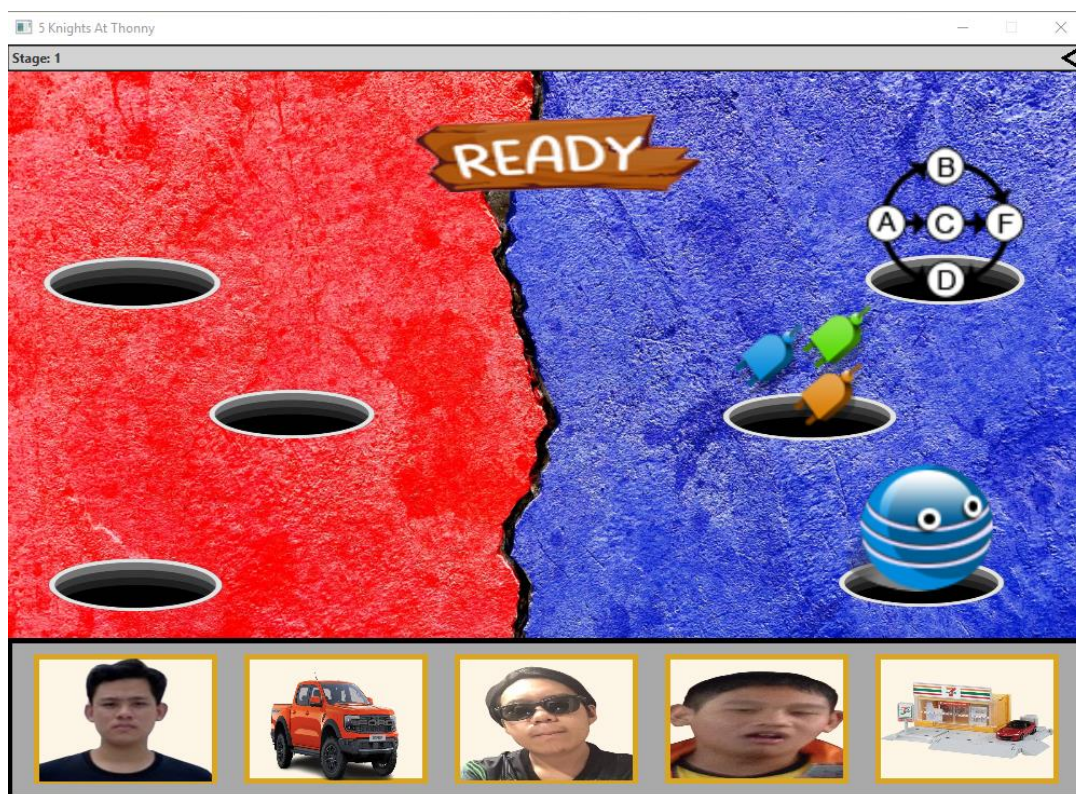
1. Tank เป็นตัวละครที่มีพลังชีวิตและโล่สูงมาก เป็นตัวที่อีกฝ่ายจะต้องตีให้ตายก่อนถึงจะเริ่มตีตัวอื่นได้
2. DPS เป็นตัวละครที่เลือดค่อนข้างน้อยแต่มีพลังโจมตีที่แรงมาก
3. Mage เป็นตัวละครที่สามารถโจมตีผ่านโล่กายภาพของตัวละครอื่นๆได้เพราะมีพลังโจมตีเป็นเวทย์แต่ถ้า มีเพียงตัวละคร Tank เท่านั้นที่มีโล่กันเวทย์
4. Healer เป็นตัวละครที่ไม่มีความโดดเด่นด้านพลังหรือความอึดมาก แต่สามารถดับเบิลคลิกเพื่อเพิ่มเลือดให้ตัวละครในสนามฝ่ายเดียวกันได้
5. Buffer เป็นตัวละครที่ไม่มีความโดดเด่นด้านพลังหรือความอึดมาก แต่สามารถดับเบิลคลิกเพื่อเพิ่มโล่ และ พลังโจมตี ให้ตัวละครในสนามฝ่ายเดียวกันได้

ไอเทม มี 5 ชนิด

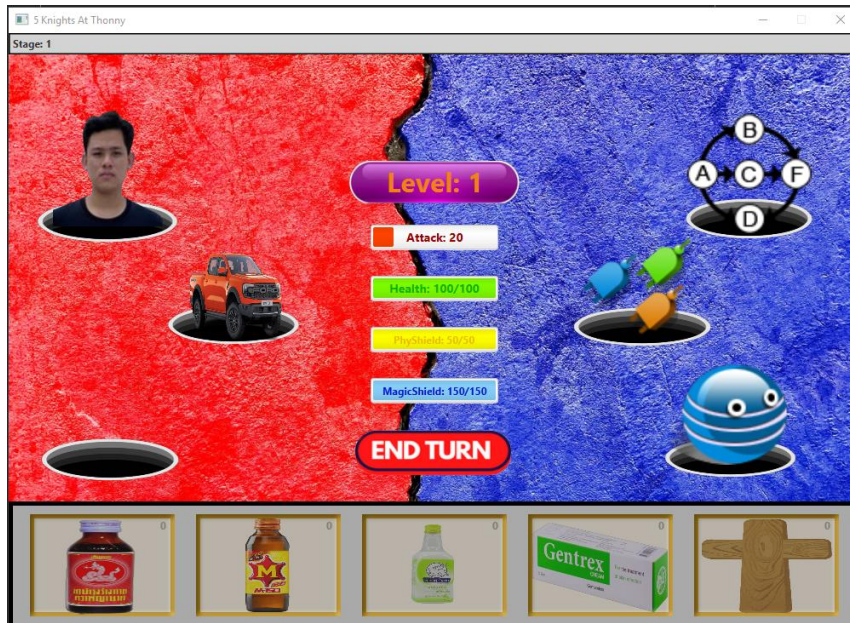
1. Heal potion เป็นไอเท็มที่สามารถเพิ่มเลือดให้ตัวละครทั้งสนามได้เมื่อกดใช้
2. Buff shield potion เป็นไอเท็มที่สามารถเพิ่มค่าโลให้ตัวละครทั้งสนามได้เมื่อกดใช้
3. Buff ATK potion เป็นไอเท็มที่สามารถเพิ่มพลังโจมตีให้ตัวละครทั้งสนามได้เมื่อกดใช้
4. Debuff potion เป็นไอเท็มที่เลือกลดโลของฝ่ายตรงข้ามได้เมื่อกดใช้
5. Revive potion เป็นไอเท็มที่เลือกชุบชีวิตตัวละครฝ่ายเราในสนามได้เมื่อกดใช้

ตัวอย่างการเล่นเกม

-เมื่อเริ่มเกมผู้เล่นจะต้องเลือกตัวละครลงสนามก่อน 3 ตัว ซึ่งตัวละครที่เลือกได้จะอยู่ในแถบด้านล่างของจอ



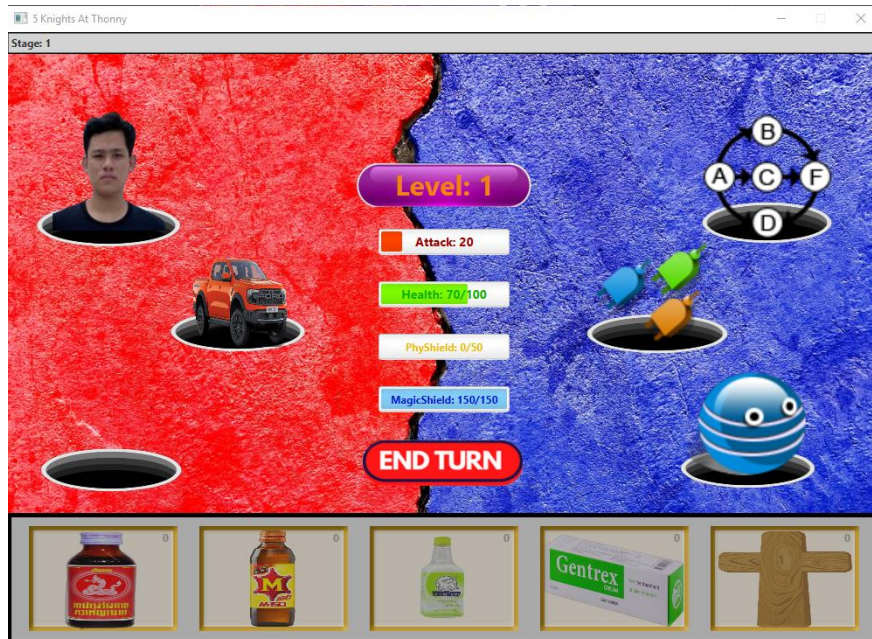
-และเมื่อเลือกตัวเสร็จก็จะเข้าเทิร์นของผู้เล่นโดยแถบด้านล่างจะถูกเปลี่ยนเป็นขวดยา ค่า stat ต่างๆของตัวละครที่เราเลือกลงมาจะแสดงอยู่ตรงกลางจอ เมื่อเราเอาเมาส์ไปวางไปบนตัวละครนั้นๆ



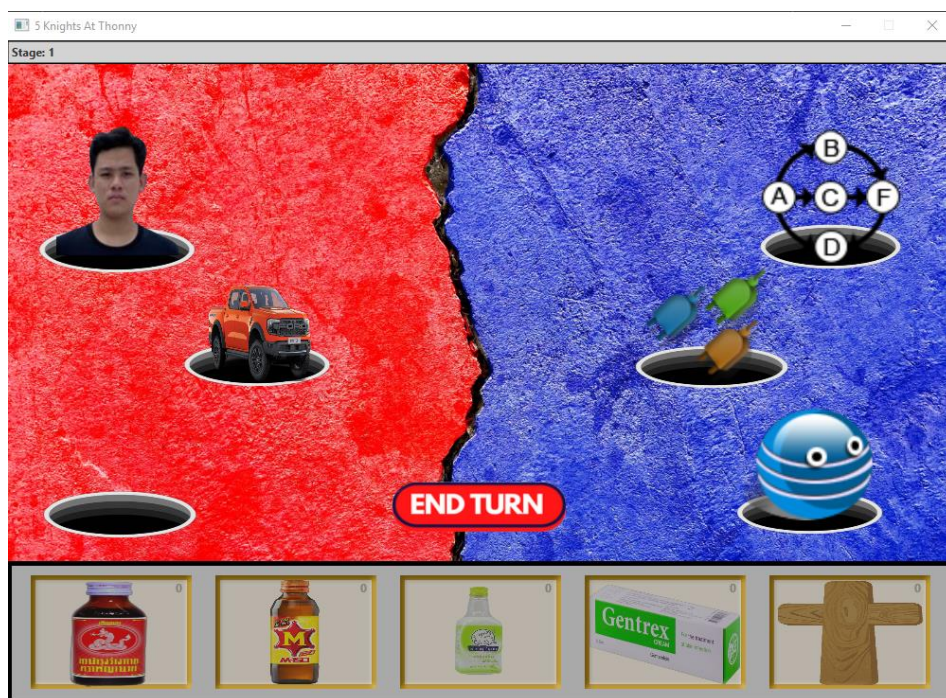
-ถ้าเราต้องการโจมตี ให้เราเอาเมาส์ไปคลิกตัวที่เป็น attacker ก่อนแล้วค่อยไปคลิกที่ defender ซึ่งจะ
ทำให้เลือดของอีกฝ่ายลดลงตามที่เห็น



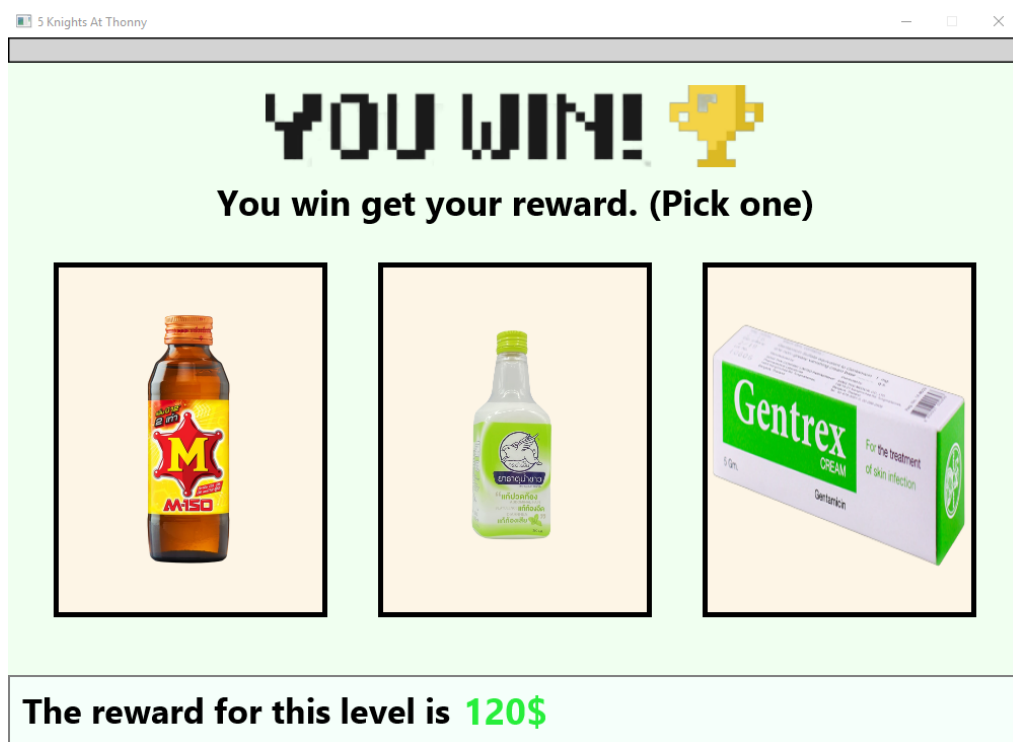
-และหลังจากเราโจมตีเสร็จให้กดปุ่ม end turn ซึ่งอีกฝ่ายจะโจมตีเรากลับและเลือดเราก็จะลดเช่นกัน



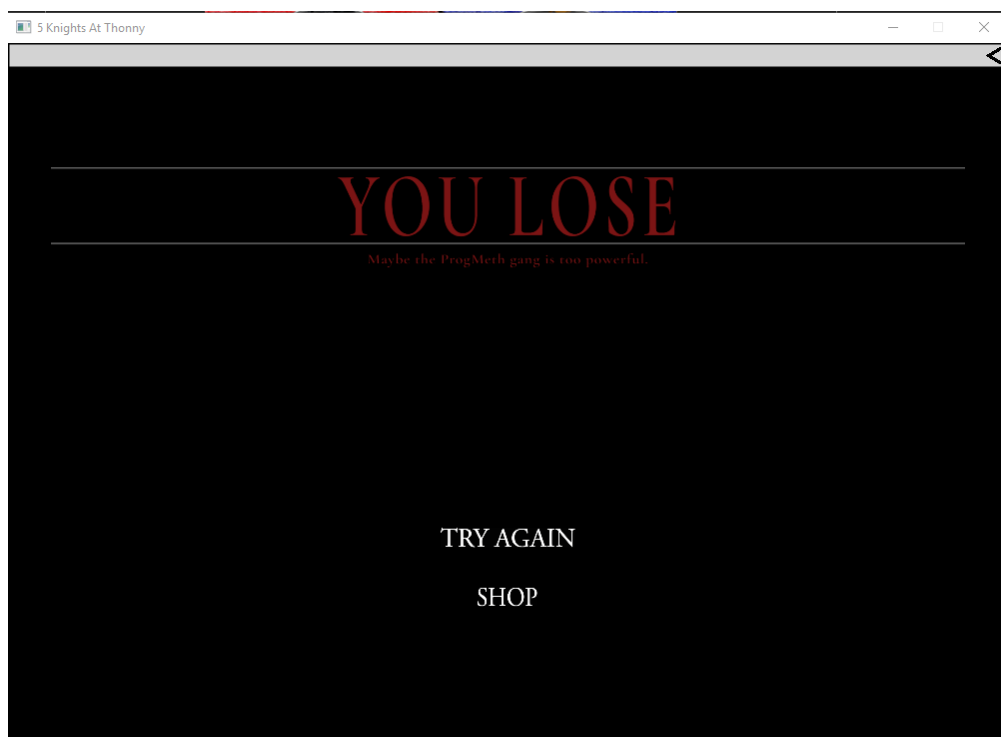
-เมื่อมีตัวใดตัวหนึ่งตาย ไอคอนของตัวละครนั้นจะมีดลงดังภาพ



-และเราจะเล่นแบบนี้ไปเรื่อย ๆ จนจบเกม ถ้าเราชนะจะมีหน้าต่างมาให้เราเลือกของรางวัลดังภาพ



-แต่ถ้าเราแพ้ก็จะมีหน้าต่างแจ้งเตือนมาดังภาพ



1.Package Application

1.1 Main = ตัวหลักของเกม

1.1.1 Methods

+ static void main(String[] args)	สั่ง launch
+ void start(Stage arg0) throws Exception	-initailize menu และสร้าง scene menu พร้อมกับเปิดหน้าต่าง menu ขึ้นมา

2.Package Controller

เป็น package ที่เก็บ class controller ที่เอาไว้ใช้ควบคุม หรือ handle สิ่งต่างๆจาก GUI ที่สร้างด้วย scene builder

2.1 Class BattleController = Controller สำหรับหน้าต่าง Battle

2.1.1 Field

- ArrayList<Interger> toAdd	ใช้เก็บตัวเลข ซึ่งกำกับถึงสิ่งที่กำลังเลือกอยู่ ณ ตอนนี ไว้ที่ Index 0 ของ ArrayList
- ArrayList<Unit> PlayerField	เก็บ Unit ของ Player
- ArrayList<Unit> EnemyField	เก็บ Unit ของ Enemy
- final String DPSURL	เก็บชื่อไฟล์ภาพของ DPS ฝั่ง Player
- final String TANKURL	เก็บชื่อไฟล์ภาพของ Tank ฝั่ง Player
- final String HEALURL	เก็บชื่อไฟล์ภาพของ Heal ฝั่ง Player
- final String BUFFURL	เก็บชื่อไฟล์ภาพของ Buff ฝั่ง Player
- final String EDPSURL	เก็บชื่อไฟล์ภาพของ DPS ฝั่ง Enemy
- final String ETANKURL	เก็บชื่อไฟล์ภาพของ Tank ฝั่ง Enemy
- final String EHEALURL	เก็บชื่อไฟล์ภาพของ Heal ฝั่ง Enemy
- final String EMAGEURL	เก็บชื่อไฟล์ภาพของ Mage ฝั่ง Enemy
- final String EBUFFURL	เก็บชื่อไฟล์ภาพของ Buff ฝั่ง Enemy

- final String HEALPOTION	เก็บชื่อไฟล์ภาพของไอเทม Heal
- final String BUFFATKPOTION	เก็บชื่อไฟล์ภาพของไอเทม BuffATK
- final String BUFFSHIELDPOTION	เก็บชื่อไฟล์ภาพของไอเทม BuffShield
- final String DEBUFFPOTION	เก็บชื่อไฟล์ภาพของไอเทม DeffBuff
- final String REVIVEPOTION	เก็บชื่อไฟล์ภาพของไอเทม Revive
- Boolean isStart	ใช้บอกว่าเกมเริ่มไปแล้ว หรือยัง
- ColorAdjust brightLight	เก็บเอฟเฟคสีขาว
- ColorAdjust blackLight	เก็บเอฟเฟคสีดำ
- Lighting lighting	ใช้สำหรับ initialize brightLight, blackLight
- Label curStage	ตัวอักษรกำกับด่าน
- ImageView hole1	เก็บรูปหลุมที่ใช้ใส่ Unit ช่องที่ 1
- ImageView hole2	เก็บรูปหลุมที่ใช้ใส่ Unit ช่องที่ 2
- ImageView hole3	เก็บรูปหลุมที่ใช้ใส่ Unit ช่องที่ 3
- ImageView slot1	เก็บรูป Unit ช่องที่ 1
- ImageView slot2	เก็บรูป Unit ช่องที่ 2
- ImageView slot3	เก็บรูป Unit ช่องที่ 3
- ImageView slot4	เก็บรูป Unit ช่องที่ 4
- ImageView slot5	เก็บรูป Unit ช่องที่ 5
- ImageView slot6	เก็บรูป Unit ช่องที่ 6
- Pane slot_pane1	เก็บ Pane ที่ใช้เก็บรูป Unit ช่องที่ 1
- Pane slot_pane2	เก็บ Pane ที่ใช้เก็บรูป Unit ช่องที่ 2
- Pane slot_pane3	เก็บ Pane ที่ใช้เก็บรูป Unit ช่องที่ 3
- Pane slot_pane4	เก็บ Pane ที่ใช้เก็บรูป Unit ช่องที่ 4
- Pane slot_pane5	เก็บ Pane ที่ใช้เก็บรูป Unit ช่องที่ 5
- Pane slot_pane6	เก็บ Pane ที่ใช้เก็บรูป Unit ช่องที่ 6
- ImageView startButton	เก็บรูปของปุ่ม start
- ImageView endButton	เก็บรูปของปุ่ม end

- ImageView back2menu	เก็บรูปของ back2menu
- ImageView level_bar	เก็บรูปของ แถบ level
- Label level	แสดง level ของ Unit
- ProgressBar atk	แสดงหลอด (เป็น %) ของ atk
- ProgressBar hp	แสดงหลอด (เป็น %) ของ hp
- ProgressBar phyShield	แสดงหลอด (เป็น %) ของ phyShield
- ProgressBar magicSheild	แสดงหลอด (เป็น %) ของ magicSheild
- Label count1	เก็บจำนวนไอเทมคงเหลือของ BuffATKPotion
- Label count2	เก็บจำนวนไอเทมคงเหลือของ BuffShieldPotion
- Label count3	เก็บจำนวนไอเทมคงเหลือของ DeBuffPotion
- Label count4	เก็บจำนวนไอเทมคงเหลือของ HealPotion
- Label count5	เก็บจำนวนไอเทมคงเหลือของ RevivePotion
- Label atkLabel	แสดงค่า atk ของ Unit
- Label hpLabel	แสดงค่า hp ของ Unit
- Label phyLabel	แสดงค่า phyLabel ของ Unit
- Label magicLabel	แสดงค่า magicLabel ของ Unit

2.1.2 Methods

+ void selectSound()	เล่นเสียงเอฟเฟค select
+ void endTurnSound()	เล่นเสียงเอฟเฟค endTurn
+ void readySound()	เล่นเสียงเอฟเฟค ready
+ void hitSound()	เล่นเสียงเอฟเฟค hit
+ void healSound()	เล่นเสียงเอฟเฟค heal
+ void buffSound()	เล่นเสียงเอฟเฟค buff
+ void Initialize()	ตั้งค่า stage level, ทำให้ปุ่ม endButton หาย, ตั้งสีตัวอักษร, วาด Enemy จาก GameLogic Initialize blackLight, brightLight
+ void draw_enemy()	วาด Enemy ทั้งหมดลงกระดาน ตาม EneyField

+ int check_Unit(Unit unit)	คืนค่า 0 หาก unit เป็น DPS, 1 หากเป็น Tank, 2 หากเป็น Heal และ 3 หากเป็น Mage นอกเหนือจากนั้นคืน 4
+ void draw(ImageView im, String URL)	set รูป ของ im ตาม URL ที่ได้รับ พร้อมปรับขนาด
+ void show_slot1()	แสดงสถานะของ Unit ของ Player ช่องแรก
+ void show_slot2()	แสดงสถานะของ Unit ของ Player ช่องสอง
+ void show_slot3()	แสดงสถานะของ Unit ของ Player ช่องสาม
+ void show_slot4()	แสดงสถานะของ Unit ของ Enemy ช่องแรก
+ void show_slot5()	แสดงสถานะของ Unit ของ Enemy ช่องสอง
+ void show_slot6()	แสดงสถานะของ Unit ของ Enemy ช่องสาม
+ void show_status(int i)	แสดงสถานะต่าง ๆ ของ Unit ที่ได้รับมาตาม index ที่รับจาก i = 0,1,2,3,4,5 แสดงถึง Unit Player ช่องแรก ไล่จนถึง Unit Enemy ช่องสุดท้ายตามลำดับ
+ void hilde_status()	ซ่อนสถานะกลางหน้าจอ
+ void toSlot1()	หากเริ่มเกมแล้วและช่องนี้ไม่มี unit จะไม่ทำอะไร ไม่เช่นนั้นจะ เคลียนเอฟเฟคทั้งหมดในหน้าจอ พร้อมเซตเอฟเฟคที่ hole เก็บ slot นี้ไว้ให้เป็น brightLight ถ้าเกมยังไม่เริ่มจะเคลียร์รูปที่ช่องนี้เก็บไว้ พร้อมทำให้ toAdd เก็บ 0 เพียงค่าเดียวพร้อมเซต PlayerField ช่อง 0 ให้เป็น null หากเกมเริ่มแล้ว และ unit นี้ยังไม่ตาย หากมีการเรียกใช้ยาเกิดขึ้น unit นี้จะได้รับผลของยานั้นยกเว้นเป็นยา Revive หากไม่มีการกดใช้ยามาก่อนหน้านี้ toAdd จะเก็บค่า 0 แทนแสดงถึง unit นี้พร้อมจะโจมตี หาก unit นี้ตายแล้ว และ ยาที่ใช้เป็น Revive Potion จะเรียกใช้ยาจากเกม Logic หากเกมเริ่มแล้ว จะมีการเรียก update_status() , reset_hole_effect() ทุกครั้ง
+ void toSlot2()	หากเริ่มเกมแล้วและช่องนี้ไม่มี unit จะไม่ทำอะไร ไม่เช่นนั้นจะ เคลียนเอฟเฟคทั้งหมดในหน้าจอ พร้อมเซตเอฟเฟคที่ hole เก็บ slot นี้ไว้ให้เป็น brightLight ถ้าเกม

	<p>ยังไม่เริ่มจะเคลียร์รูปที่ช่องนี้เก็บไว้ พร้อมทำให้ toAdd เก็บ 1 เพียงค่าเดียวพร้อมเซต PlayerField ช่อง 1 ให้เป็น null หากเกมเริ่มแล้ว และ unit นี้ยังไม่ตาย หากมีการเรียกใช้ยาเกิดขึ้น unit นี้จะได้รับผลของยานั้นยกเว้นเป็นยา Revive หากไม่มีการกดใช้ยามาก่อนหน้านี้ toAdd จะเก็บค่า 1 แทนแสดงถึง unit นี้พร้อมจะโจมตี หาก unit นี้ตายแล้ว และ ยาที่ใช้เป็น Revive Potion จะเรียกใช้ยาจากเกม Logic หากเกมเริ่มแล้ว จะมีการเรียก update_status() , reset_hole_effect() ทุกครั้ง</p>
+ void toSlot3()	<p>หากเริ่มเกมแล้วและช่องนี้ไม่มี unit จะไม่ทำอะไร ไม่เช่นนั้นจะ เคลียนเอฟเฟคทั้งหมดในหน้าจอ พร้อมเซตเอฟเฟคที่ hole เก็บ slot นี้ไว้ให้เป็น brightLight ถ้าเกมยังไม่เริ่มจะเคลียร์รูปที่ช่องนี้เก็บไว้ พร้อมทำให้ toAdd เก็บ 2 เพียงค่าเดียวพร้อมเซต PlayerField ช่อง 2ให้เป็น null หากเกมเริ่มแล้ว และ unit นี้ยังไม่ตาย หากมีการเรียกใช้ยาเกิดขึ้น unit นี้จะได้รับผลของยานั้นยกเว้นเป็นยา Revive หากไม่มีการกดใช้ยามาก่อนหน้านี้ toAdd จะเก็บค่า 2 แทนแสดงถึง unit นี้พร้อมจะโจมตี หาก unit นี้ตายแล้ว และ ยาที่ใช้เป็น Revive Potion จะเรียกใช้ยาจากเกม Logic หากเกมเริ่มแล้ว จะมีการเรียก update_status() , reset_hole_effect() ทุกครั้ง</p>
+ void reset_hole_effect()	<p>เคลีย effect ของหลุมใส่ Unit ผัง Player ทั้งหมด</p>
+ void reset_pane_effect()	<p>เคลีย effect ของ pane ที่ใช้เก็บรูปของ Item ทั้งหมด</p>
+ void addBar1()	<p>หากเกมยังไม่เริ่ม และ มีการเลือกช่องที่จะใส่ unit นั้นแล้ว เพิ่ม Tank ลงช่องนั้นแล้วยกเลิกปุ่มนี้ หากเกมเริ่มไปแล้วจะเป็นการเตรียมใช้ไอเทม BuffATK แทน และเซตเอฟเฟคว่ามีการกดใช้งาน</p>
+ void addBar2()	<p>หากเกมยังไม่เริ่ม และ มีการเลือกช่องที่จะใส่ unit นั้นแล้ว เพิ่ม Heal ลงช่องนั้นแล้วยกเลิกปุ่มนี้ หากเกมเริ่มไป</p>

	แล้วจะเป็นการเตรียมใช้ไอเทม BuffShield แทน และเช็คเอฟเฟคว่ามีการกดใช้งาน
+ void addBar3()	หากเกมยังไม่เริ่ม และ มีการเลือกช่องที่จะใส่ unit นั้น แล้ว เพิ่ม Heal ลงช่องนั้นแล้วยกเลิกปุ่มนี้ หากเกมเริ่มไปแล้วจะเป็นการเตรียมใช้ไอเทม DeBuff แทน และเช็คเอฟเฟคว่ามีการกดใช้งาน
+ void addBar4()	หากเกมยังไม่เริ่ม และ มีการเลือกช่องที่จะใส่ unit นั้น แล้ว เพิ่ม Mage ลงช่องนั้นแล้วยกเลิกปุ่มนี้ หากเกมเริ่มไปแล้วจะเป็นการเตรียมใช้ไอเทม Heal แทน และเช็คเอฟเฟคว่ามีการกดใช้งาน
+ void addBar5()	หากเกมยังไม่เริ่ม และ มีการเลือกช่องที่จะใส่ unit นั้น แล้ว เพิ่ม Buff ลงช่องนั้นแล้วยกเลิกปุ่มนี้ หากเกมเริ่มไปแล้วจะเป็นการเตรียมใช้ไอเทม Revive แทน และเช็คเอฟเฟคว่ามีการกดใช้งาน
+ void reset_pane(Unit unit)	ใช้ยกเลิกผลที่สั่งให้ pane ที่เก็บ unit นั้นใช้งานไม่ได้
+ void start_battle()	กดเมื่อพร้อมเข้าหน้าต่อสู้ หาก Player ไม่ได้ใส่ Unit จะไม่ทำอะไร พร้อมขึ้น Alert เตือนให้ใส่ Unit อย่างน้อย 1 ตัว ถ้า Player Unit มี Unit อย่างน้อย 1 ตัว จะเรียก readySound() พร้อม set ไม่ให้กลับไปหน้า menu, ตั้งค่า isStart ให้เป็น true ลบปุ่ม startButton ทำให้ pane กลับมาใช้งานได้ ตั้งค่า count ของ Item, ส่ง PlayerField ไปให้ GameLogic, เคลียร์ effect ทั้งหมดของ hole , เรียก draw_items() และเรียก update_status()
+ void update_status()	เช็คสถานะของ Unit และ Items ทั้งหมด หาก Unit ตาย จะ ตั้ง effect ของ slot ที่ใส่ Unit นั้นให้เป็น blackLight หากเป็น Item ก็จะ ตั้งค่า count ใหม่ และหากไอเทมนั้นหมด ก็จะตั้ง effect blackLight ให้ pane ที่ใส่ Item นั้น
+ draw_items()	วาด bar ใหม่ให้เป็นรูป item ตามลำดับ

+atkE1()	หากเกมยังไม่เริ่ม หรือ ไม่มี Unit ที่หมายจะตี จะไม่ทำอะไร ไม่เช่นนั้นจะเรียก hitSound() หากเป็น unit Player ที่ action มาและใช้ GameLogic คำนวณความเสียหายที่เกิดกับ Enemy Unit ตัวแรก พร้อมเรียก update_status(), game_end() และ show_status(3)
+atkE2()	หากเกมยังไม่เริ่ม หรือ ไม่มี Unit ที่หมายจะตี จะไม่ทำอะไร ไม่เช่นนั้นจะเรียก hitSound() หากเป็น unit Player ที่ action มาและใช้ GameLogic คำนวณความเสียหายที่เกิดกับ Enemy Unit ตัวสอง พร้อมเรียก update_status(), game_end() และ show_status(4)
+atkE3 ()	หากเกมยังไม่เริ่ม หรือ ไม่มี Unit ที่หมายจะตี จะไม่ทำอะไร ไม่เช่นนั้นจะเรียก hitSound() หากเป็น unit Player ที่ action มาและใช้ GameLogic คำนวณความเสียหายที่เกิดกับ Enemy Unit ตัวสามพร้อมเรียก update_status(), game_end() และ show_status(5)
+ enemyAttack()	เรียก endTurnSound() เชื้อตให้ slot ทั้งหมดกลับมาใช้งานได้ เรียก reset_pane_effect(), reset_hole_effect(), คำนวณศัตรูให้ตีจาก GameLogic.enemyAttack()เคลีย toAdd, เรียก update_status และ เรียก game_end()
+ game_end()	หากเกมยังไม่จบจะไม่ทำอะไร หากเกมจบแล้ว หาก Player เป็นฝ่ายชนะ จะเปลี่ยนหน้าไปหน้า Reward ไม่เช่นนั้น จะเปลี่ยนไปหน้า Lose

2.2 Class MenuController = Controller สำหรับหน้าต่าง Menu

2.2.1 Field

- ImageView startGameButton	ปุ่มชื่อ Start Game ที่เอาไว้กดเพื่อ start game
--------------------------------	---

- ImageView shopButton	ปุ่มชื่อ Shop ที่เอาไว้กดเพื่อเข้าไปหน้า Shop
- ImageView exitGameButton	ปุ่มชื่อ Exit Game ที่เอาไว้กดเพื่อปิดหน้าต่าง Game
- String NEWGAMEURL	เก็บชื่อรูปปุ่ม New Game
- String CONTINUEURL	เก็บชื่อรูปปุ่ม Continue
- String SHOPURL	เก็บชื่อรูปปุ่ม Shop
- String HELPURL	เก็บชื่อรูปปุ่ม Help
- String EXITURL	เก็บชื่อรูปปุ่ม Exit
- String NEWGAMESLEURL	เก็บชื่อรูปปุ่ม New Game เมื่อมีเม้าส์มาวาง
- String CONTINUNESLEURL	เก็บชื่อรูปปุ่ม Continue เมื่อมีเม้าส์มาวาง
- String SHOPSLEURL	เก็บชื่อรูปปุ่ม Shop เมื่อมีเม้าส์มาวาง
- String HELPSLEURL	เก็บชื่อรูปปุ่ม Help เมื่อมีเม้าส์มาวาง
- String EXITUSLERL	เก็บชื่อรูปปุ่ม Exit เมื่อมีเม้าส์มาวาง
- Boolean isStart	คืนค่าว่าเกมเริ่มไปแล้วหรือยัง
- ColorAdjust blackLight	เก็บเอฟเฟค blackLight
- Lighting lighting	เอาไว้ initialize blackLight
- MediaPlayer mediaPlayer	เก็บเสียง Background

2.2.2 Methods

+ void initialize(URL arg0, ResourceBundle arg1)	เป็น function ที่จะถูกเรียกใช้งานเมื่อเราเปิดมาที่หน้าต่าง Menu แต่ใน class นี้เราไม่ได้ให้มันทำอะไร
+ void startGameButtonHandler()	- สร้าง enemy level 1 หลังจากกดปุ่ม start game เพื่อให้เกมพร้อมเล่น - เปลี่ยนจากหน้าต่าง Menu เป็น Battle
+ void shopButtonHandler()	- เปลี่ยนจากหน้าต่าง Menu เป็น Shop
+ void helpButtonHandler()	- เปลี่ยนจากหน้า Menu เป็น Help
+ void exitButtonHandler()	- ปิดหน้าต่างที่เล่นเกมอยู่
+ void bgSound()	เล่นเสียงฉากหลัง

+ void selectSound()	เล่นเสียงเลือก
+ void mouseEnter1	เรียก mouseEnter(0)
+ void mouseEnter2	เรียก mouseEnter(1)
+ void mouseEnter3	เรียก mouseEnter(2)
+ void mouseEnter4	เรียก mouseEnter(3)
+ void mouseEnter5	เรียก mouseEnter(4)
+ void draw(ImageView im, String URL)	วาด im ด้วย URL ที่ได้รับพร้อมปรับขนาด
+ void mouseEnter(int i)	วาดรูป New Game, Continue, Shop, Help, Exit ใหม่ตาม I ที่ได้รับเรียงจาก 0 ไป 4 ตามลำดับ
+ void mouseExit()	วาดรูปปุ่มเป็นตั้งต้นใหม่ให้ทุกปุ่ม
+ void continueButtonHandler()	เรียก selectSound() พร้อมเปลี่ยนไปหน้า battle
+ void updateContinueButton()	ถ้าเล่นเกมเคลี่ยนไปแล้ว 1 ด้าน ปุ่ม continue จะสามารถใช้ได้
+ void newGameButtonHandler()	เรียก selectSound() พร้อมเปลี่ยนไปหน้า battle พร้อมทำ alert หาก continue ใช้ งานได้ และกำหนด level ตาม GameLogic

2.3 Class ShopController = Controller สำหรับหน้าต่าง Shop

2.3.1 Field

- Text moneyText	เป็น text ที่แสดงยอดเงินคงเหลือของผู้เล่น
- Text tankText	เป็น text ที่บอกว่าเลเวลถัดไปของตัวละคร Tank เราคืออะไร
- Text DPSText	เป็น text ที่บอกว่าเลเวลถัดไปของตัวละคร DPS เราคืออะไร

- Text mageText	เป็น text ที่บอกว่าเลเวลถัดไปของตัวละคร Mage เราคืออะไร
- Text healerText	เป็น text ที่บอกว่าเลเวลถัดไปของตัวละคร Healer เราคืออะไร
- Text bufferText	เป็น text ที่บอกว่าเลเวลถัดไปของตัวละคร Buffer เราคืออะไร
- ImageView treasureButton	เป็นปุ่มสำหรับกดซื้อหีบสุม Item
- Label tankButton	เป็นปุ่มสำหรับกดอัฟเลเวล Tank
- Label DPSButton	เป็นปุ่มสำหรับกดอัฟเลเวล DPS
- Label mageButton	เป็นปุ่มสำหรับกดอัฟเลเวล Mage
- Label healerButton	เป็นปุ่มสำหรับกดอัฟเลเวล Healer
- Label bufferButton	เป็นปุ่มสำหรับกดอัฟเลเวล Buffer
- ImageView returnButton	เป็นปุ่มสำหรับกดเพื่อ กลับไปหน้า menu

2.3.2 Methods

+ void initialize(URL arg0, ResourceBundle arg1)	เป็น function ที่จะถูกเรียกใช้งานเมื่อเราเปิดมาที่หน้าต่าง Shop ซึ่งเราจะทำการเรียก function ทั้งหมดที่ใช้สำหรับการ Update หน้าต่าง Shop
+ void updateMoneyText()	อัปเดตยอดเงินคงเหลือของผู้เล่น
+ void updateTankText()	อัปเดตเลเวลถัดไปของตัวละคร Tank
+ void updateDPSText()	อัปเดตเลเวลถัดไปของตัวละคร DPS
+ void updateMageText()	อัปเดตเลเวลถัดไปของตัวละคร Mage
+ void updateHealerText()	อัปเดตเลเวลถัดไปของตัวละคร Healer
+ void updateBufferText()	อัปเดตเลเวลถัดไปของตัวละคร Buffer
+ void updateTankButton()	ปรับราคาของการอัฟเลเวล Tank ให้เท่ากับเลเวลของมันคูณ 10

+ void updateDPSButton()	ปรับราคาของการอัปเลเวล DPS ให้เท่ากับเลเวลของม้านicorn 10
+ void updateMageButton()	ปรับราคาของการอัปเลเวล Mage ให้เท่ากับเลเวลของม้านicorn 10
+ void updateHealerButton()	ปรับราคาของการอัปเลเวล Healer ให้เท่ากับเลเวลของม้านicorn 10
+ void updateBufferButton()	ปรับราคาของการอัปเลเวล Buffer ให้เท่ากับเลเวลของม้านicorn 10
+ void treasureHandler()	สุ่มของจาก Item ทั้งหมดในเกมแล้วแสดงผลบอกผู้เล่นด้วยการเรียก ฟังก์ชัน alertTreasure() และเรียก purchaseSound()
+ void purchaseSound()	เล่นเสียงซื้อ
+ void selectSound()	เล่นเสียงเลือก
+ void tankLevelUpHandler()	อัปเลเวลให้ตัวละคร Tank และเรียกใช้ฟังก์ชันอัปเดต text และ button ของ Tank และเรียก purchaseSound()
+ void DPSLevelUpHandler()	อัปเลเวลให้ตัวละคร DPS และเรียกใช้ฟังก์ชันอัปเดต text และ button ของ DPS และเรียก purchaseSound()
+ void mageLevelUpHandler()	อัปเลเวลให้ตัวละคร Mage และเรียกใช้ฟังก์ชันอัปเดต text และ button ของ Mage และเรียก purchaseSound()
+ void healerLevelUpHandler()	อัปเลเวลให้ตัวละคร Healer และเรียกใช้ฟังก์ชันอัปเดต text และ button ของ Healer และเรียก purchaseSound()
+ void bufferLevelUpHandler()	อัปเลเวลให้ตัวละคร Buffer และเรียกใช้ฟังก์ชันอัปเดต text และ button ของ Buffer และเรียก purchaseSound()

+ void alertTreasure()	แสดง alert ที่บอกว่าเราได้อะไรจากการสุ่มของใน treasure
+ void alertNotEnoughMoney()	แสดง alert เมื่อจำนวนเงินของเราไม่พอต่อการซื้อของ
+ void returnButtonHandler()	เป็น handler ที่จะเปลี่ยนหน้าต่างให้เป็นหน้า menu (เพื่อออกจากหน้า shop) และเรียก selectSound()

2.4 Class HelpController = Controller สำหรับหน้าต่าง Help

2.4.1 Field

- ImageView back2menu	เก็บรูปกลับไปหน้าเมนู
-----------------------	-----------------------

2.4.2 Methods

+ void selectSound()	เล่นเสียงเลือก
+ void back2menuHandler()	selectSound() พร้อมกลับไปหน้าเมนู

2.5 Class LoseController = Controller สำหรับหน้าต่าง Lose

2.5.1 Field

- final String TRYAGAINURL	เก็บชื่อรูป TryAgain
- final String SHOPURL	เก็บชื่อรูป Shop
- final String TRYAGAINSLEURL	เก็บชื่อรูป TryAgain เมื่อมีเม้าส์มาวาง
- final String SHOPSLEURL	เก็บชื่อรูป Shop เมื่อมีเม้าส์มาวาง
- ImageView back2menu	เก็บรูปกลับไปหน้าเมนู
- ImageView TryAgian	เก็บรูป TryAgain
- ImageView Shop	เก็บรูป Shop

2.5.2 Methods

+ void selectSound()	เล่นเสียงเลือก
----------------------	----------------

+ void loseSound()	เล่นเสียงแพ้
+ void initialize(URL arg0,ResourceBundle arg1)	เรียก loseSound()
+ void mouseEnter1()	เรียก mouseEnter(0)
+ void mouseEnter2()	เรียก mouseEnter(1)
+ void draw(Imageview im,String URL)	set รูป ของ im ตาม URL ที่ได้รับ พร้อมปรับขนาด
+ void mouseEnter(int i)	วาด TryAgain และ Shop ใหม่ให้เป็นที่เลือก ตามค่า i = 0, 1 ตามลำดับ
+ void mouseExit()	วาด TryAgain และ Shop ใหม่ให้เป็นที่ค่าตั้งต้น
+ void back2menu() throws Exception	เรียก selectSound() และกลับไปหน้า menu
+ void returnButtonHandler() throws Exception	เรียก selectSound() และกลับไปหน้า Battle โดยสร้าง Enemy ใหม่ตาม level โดยใช้ GameLogic
+ void go2shop() throws Exception	เรียก selectSound() และไปหน้า Shop

2.6 Class RewardController = Controller สำหรับหน้าต่าง Reward

2.6.1 Field

- boolean hasPicked	คิดว่าได้เลือกรางวัลไปแล้วหรือไม่
+ int item1	เก็บค่า int แสดงถึงชนิดไอเทมแต่ละอย่าง
+ int item2	เก็บค่า int แสดงถึงชนิดไอเทมแต่ละอย่าง
+ int item3	เก็บค่า int แสดงถึงชนิดไอเทมแต่ละอย่าง
- final String HEALPOTION	เก็บชื่อรูป HealPotion
- final String BUFFATKPOTION	เก็บชื่อรูป BuffATKPotion
- final String BUFFSHIELDPOTION	เก็บชื่อรูป BuffShieldPotion
- final String DEBUFFPOTION	เก็บชื่อรูป DeBuffPotion
- final String REVIVEPOTION	เก็บชื่อรูป RevivePotion
- String URL	เอาไว้เก็บชื่อรูปที่ต้องการวาด
- Text rewardMoneyText	ตัวอักษรเก็บเงินที่ได้เมื่อเคลียด่านไปแล้ว
- Text topicText	ตัวอักษรแสดงข้อความ

- ImageView itemImage1	เก็บรูป item ช่อง 1
- ImageView itemImage2	เก็บรูป item ช่อง 2
- ImageView itemImage3	เก็บรูป item ช่อง 3
- ImageView back2menuButton	เก็บรูป กลับไปหน้าเมนู
- ImageView continueButton	เก็บรูป continue
- ColorAdjust blackLight	เก็บเอฟเฟคสีดำ
- Lighting lightning	เอาไว้ initialize blackLight

2.6.2 Methods

+ void chooseSound()	เล่นเสียง เลือก Item
+ void selectySound()	เล่นเสียง เลือก
+ void winSound()	เล่นเสียง ชนะ
+ void initialize(URL arg0,ResourceBundle arg1)	เรียก winSound(), draw_all() รับ reward จาก GameLogic,
+ int updateMoneyText()	แก้ค่าเงินที่ได้เมื่อเคลียร์ด่าน โดยเรียกฟังก์ชัน จาก GameLogic
+ void randomItem()	สุ่ม Item ขึ้นที่ 1, 2, 3
+ void draw_all()	วาดรูป Item ทั้ง 3 ตามที่ได้สุ่มไว้
+ void draw(ImageView im,int i)	วาดรูป im ตามค่า i ที่ได้รับจาก BuffATK, BuffShield, Debuff, Heal, Revive ตามค่า i = 0, 1, 2, 3, 4 ตามลำดับ และปรับขนาดรูป
+ void add_black_effect()	Set เอฟเฟคให้ทุกรูป item เป็น blackLight
+ void Image1Handler()	หาก เลือกไอเทมไปแล้วจะไม่ทำอะไร ถ้ายังไม่เลือกจะเลือกไอเทมขึ้นที่ 1 set blackLight ให้ทุกรูปยกเว้นรูปที่ 1 , เรียก chooseSound(), Add Item ไปให้ผู้เล่นโดยใช้ GameLogic, แก้ค่า text ที่แสดง และ แก้ hasPicked ให้เป็น true
+ void Image2Handler()	หาก เลือกไอเทมไปแล้วจะไม่ทำอะไร ถ้ายังไม่เลือกจะเลือกไอเทมขึ้นที่ 2 set blackLight ให้ทุกรูปยกเว้นรูปที่ 2,

	เรียก chooseSound(), Add Item ไปให้ผู้เล่นโดยใช้ GameLogic, แก่ค่า text ที่แสดง และ แก่ hasPicked ให้เป็น true
+ void Image3Handler()	หาก เลือกไอเทมไปแล้วจะไม่ทำอะไร ถ้ายังไม่เลือกจะเลือกไอเทมชิ้นที่ 3 set blackLight ให้ทุกรูปยกเว้นรูปที่ 3 , เรียก chooseSound(), Add Item ไปให้ผู้เล่นโดยใช้ GameLogic, แก่ค่า text ที่แสดง และ แก่ hasPicked ให้เป็น true
+ void updateTopicText(String text)	หากยังไม่เลือกไอเทม topic จะเป็น text แสดงว่า "You win get your reward. (Pick one)" ถ้าเลือกแล้ว จะ แสดง text ตาม พารามิเตอร์ ที่ได้รับ
+ void back2menuButtonHandler() throws Exception	เรียก selectSound() พร้อมกลับไปหน้า Menu
+ void continueButtonHandler() throws Exception	ถ้ายังไม่เลือกจะไม่ทำอะไร ถ้าเลือกแล้ว จะเรียก selectSound() พร้อมไปหน้า Battle

3.Package entity.base

เป็น package ที่เก็บ class พื้นฐานของเกมนี้ทั้งหมด ทั้ง item , unit และรวมถึง interface ต่างๆที่ต้องใช้ เพื่อนที่จะไปสร้าง Item หรือตัวละครจริงๆได้โดย extends และ implements เอาไม่ต้องสร้างใหม่ทุก class

3.1 Abstract class Unit = เป็นคลาสแม่ของ Unit ทุกตัวในเกมนี้

2.1.1 Field

# String name	เก็บชื่อของตัวละคร
# int maxHealth	เก็บค่าพลังชีวิตที่มากที่สุด โดยมีค่าเริ่มต้นเป็น 100
# int health	เก็บค่าพลังชีวิตที่มีในปัจจุบัน โดยมีค่าเริ่มต้นเป็น 100

# int basicAttack	เก็บค่าพลังโจมตีพื้นฐานของตัวละคร โดยมีค่าเริ่มต้นเป็น 20
# int attack	เก็บค่าพลังโจมตี โดยมีค่าเริ่มต้นเป็น 20
# int maxShieldPhy	เก็บค่าโล่กายภาพที่มากที่สุด โดยมีค่าเริ่มต้นเป็น 50
# int shieldPhy	เก็บค่าโล่กายภาพที่มีในปัจจุบัน โดยมีค่าเริ่มต้นเป็น 50
# int level	เก็บค่าเลเวลของตัวละคร โดยมีค่าเริ่มต้นเป็น 1
# Boolean isAlive	เก็บค่า Boolean ว่าตัวละครนี้ยังมีชีวิตอยู่หรือไม่ โดยมีค่าเริ่มต้นเป็น true
# String URL	เก็บ URL ของภาพของตัวละครนั้นๆ

2.1.2 Constructor

+ Unit(String name, String URL)	รับชื่อและ URL ของภาพ เพื่อสร้าง Unit ใหม่ ส่วนของ field ที่เหลือที่การกำหนดค่าเริ่มต้นไว้หมดแล้ว
---------------------------------	---

2.1.3 Methods

+ abstract void levelUp()	อัปเดต stat ต่างๆของตัวละครเมื่ออัปเดตเลเวล
+ abstract void reset()	Reset เลือด โล่ และค่าพลังโจมตีของตัวละครเมื่อจบเกม เพื่อให้พร้อมสำหรับเล่นตาใหม่
+ String getName()	getter ของ field name
+ void setName(String Name)	Setter ของ field name
+ int getMaxHealth()	getter ของ field maxHealth
+ void setMaxHealth(int maxHealth)	Setter ของ field maxHealth ซึ่งค่าของ maxHealth ห้ามต่ำกว่า 1

+ int getHealth()	getter ของ field health
+ void setHealth(int health)	Setter ของ field health
+ int getAttack()	getter ของ field attack
+ void setAttack(int attack)	Setter ของ field attack ซึ่งค่า Attack ที่รับเข้ามาถ้าต่ำกว่า 0 จะถูกปรับเป็น 1
+ int getShieldPhy()	getter ของ field ShieldPhy
+ void setShieldPhy(int shieldPhy)	Setter ของ field ShieldPhy ซึ่งถ้าค่า shieldPhy ที่รับมาน้อยกว่า 0 จะถูกเปลี่ยนเป็น 0
+ int geMaxtShieldPhy()	getter ของ field maxShieldPhy
+ void setMaxShieldPhy(int maxShieldPhy)	Setter ของ field maxShieldPhy
+ boolean isAlive()	getter ของ field isAlive
+ void setAlive(boolean isAlive)	Setter ของ field isAlive
+ int getLevel()	getter ของ field level
+ void setLevel(int level)	Setter ของ field level ซึ่งจะทำการอัปเดตเลเวลของตัวละครให้ตรงกับเลเวลที่รับมาด้วย
+ String getURL()	getter ของ field URL
+ void setURL(String URL)	Setter ของ field URL
+ int getBasicAttack()	getter ของ field basicAttack
+ void setBasicAttack(int basicAttack)	Setter ของ field basicAttack
+ void decreaseHealth(int damage)	ลดเลือดของตัวละครตาม damage ที่รับเข้ามาและถ้าตัวละครเหลือเลือดไม่มากกว่า 0 Field isAlive จะถูกเปลี่ยนเป็น false เพื่อแสดงว่า Unit นี้ตายแล้ว
+ int decreaseShieldPhy(int damage)	ลดเกราะของตัวละครตาม damage ที่ได้รับมาและคืนค่า int ที่บอกว่าdamage หลังหักลบเกราะออกไปแล้วเหลือเท่าไร เพื่อจะไปลดเลือดต่อ

3.2 Abstract class Items = เป็นคลาสแม่ของ Items ทุกชิ้นในเกมนี้

3.2.1 Field

# String name	เก็บชื่อของ item
# int count	เก็บจำนวนที่มีของ item ชิ้นนี้
# String URL	เก็บ URL ของรูปภาพของ Item ชิ้นนี้

3.2.2 Constructor

+ Items(String name)	รับค่า name และ set ค่าตามที่ได้รับมา และ set count ไว้เป็น 0
----------------------	---

3.2.3 Methods

+ int getCount()	getter ของ field count
+ void setCount(int count)	Setter ของ field count ซึ่งค่า count จะมากกว่าเท่ากับ 0 เสมอ
+ String getURL()	getter ของ field URL
+void setURL(String URL)	Setter ของ field URL
+ String getName()	getter ของ field name
+ void setName(String name)	Setter ของ field name

3.3 Interface BuffATKable = เป็น interface ที่บอกว่าเราสามารถเพิ่มพลังโจมตีให้ Unit ได้

3.3.1 Methods

+ void buffATK(Unit e)	เป็น function ที่จะทำการเพิ่มพลังโจมตีให้ unit ที่รับมา (ต้องไป implements ใน class ที่ implements interface นี้ไป)
------------------------	---

3.4 Interface BuffShieldable = เป็น interface ที่บอกว่าเราสามารถเพิ่มShieldให้ Unit ได้

3.4.1 Methods

+ void buffShield(Unit e)	เป็น function ที่จะทำการเพิ่มShield ให้ unit ที่รับมา (ต้องไป implements ใน class ที่ implements interface นี้ไป)
---------------------------	---

3.5 Interface deBuffable = เป็น interface ที่บอกว่าเราสามารถลด Shield ให้ Unit ได้

3.5.1 Methods

+ void debuff(Unit e)	เป็น function ที่จะทำการลดShield ของ unit ที่รับมา (ต้องไป implements ใน class ที่ implements interface นี้ไป)
-----------------------	--

3.6 Interface Healable = เป็น interface ที่บอกว่าเราสามารถเพิ่ม Health ให้ Unit ได้

3.6.1 Methods

+ void heal(Unit e)	เป็น function ที่จะทำการเพิ่ม health ของ unit ที่รับมา (ต้องไป implements ใน class ที่ implements interface นี้ไป)
---------------------	--

3.7 Interface Reviveable = เป็น interface ที่บอกว่าเราสามารถชุบชีวิตให้ Unit ได้

3.7.1 Methods

+ void revive(Unit e)	เป็น function ที่จะทำการชุบชีวิตให้ unit ที่รับมา (ต้องไป implements ใน class ที่ implements interface นี้ไป)
-----------------------	---

3.8 Interface Levelable = เป็น interface ที่บอกว่าเรามี level และสามารถอัป level ได้

3.8.1 Methods

+ void levelUp()	เป็น function ที่จะทำการอัป level และเพิ่มค่า stat ต่างๆให้ unit ที่รับมา (ต้องไป implements ใน class ที่ implements interface นี้ไป)
------------------	---

4.Package entity.item

เป็น package ที่เก็บ class ของ item ทั้งหมดที่มีในเกมนี้ โดยทุกๆ class จะ extends มาจาก class Items จาก entity.base

4.1 Class BuffATKPotion = ขวดยาเพิ่มพลังโจมตี ซึ่ง extends มาจาก Items

4.1.1 Field

+ int BuffATK	เป็นค่าที่แสดงถึงพลังโจมตีที่เราสามารถเพิ่มให้ Unit อื่นได้
---------------	---

4.1.2 Constructor

+ BuffATKPotion()	เรียก constructor ของ Item โดยใช้ชื่อของ Potion เข้าไปและ Set ค่า URL กับ BuffATK โดยมีค่าเริ่มต้นที่ 30
-------------------	--

4.1.3 Methods

+ void buffATK(Unit e)	เพิ่มพลังโจมตีให้ Unit ที่ถูกเลือกและลดจำนวนที่มีของ potion ชนิดนี้ลง
------------------------	---

4.2 Class BuffShieldPotion = ขวดยาเพิ่มโล่ ซึ่ง extends มาจาก Items

4.2.1 Field

+ int buffShield	เป็นค่าที่แสดงถึงโล่ที่เราสามารถเพิ่มให้ Unit อื่นได้
------------------	---

4.2.2 Constructor

+ BuffShieldPotion()	เรียก constructor ของ Item โดยใช้ชื่อของ Potion เข้าไปและ Set ค่า URL กับ buffShield โดยมีค่าเริ่มต้นที่ 30
----------------------	---

4.2.3 Methods

+ buffShield(Unit e)	เพิ่มโล่ให้ Unit ที่ถูกเลือก โดยถ้า unit ที่เลือกเป็น Tank จะเพิ่มทั้งเกราะ physical และ Magic และการเพิ่มโล่นั้นไม่สามารถเพิ่มจน
----------------------	---

	เกินค่า maxShield ได้ และหลังจากเพิ่มเสร็จ จะทำการลดจำนวนของ potion ชนิดนี้ลง
--	---

4.3 Class deBuffPotion = ขวดยาลดโล่ ซึ่ง extends มาจาก Items

4.3.1 Field

+ int debuffShield	เป็นค่าที่แสดงถึงปริมาณของโล่ที่ potion นี้ สามารถลดได้
--------------------	---

4.3.2 Constructor

+ debuffPotion()	เรียก constructor ของ Item โดยใช้ชื่อของ Potion เข้าไปและ Set ค่า URL กับ debuffShield โดยมีค่าเริ่มต้นที่ 50
------------------	---

4.3.3 Methods

+ void debuff(Unit e)	ลดโล่ของ Unit ที่ถูกเลือกโดยถ้า Unit นั้น เป็น tank จะถูกลดทั้ง physical และ magic shield หลังจากลดเกราะเสร็จจะทำการลด จำนวนที่มีของ potion ชนิดนี้ลง
-----------------------	---

4.4 Class HealPotion = ขวดยาเพิ่มเลือด ซึ่ง extends มาจาก Items

4.4.1 Field

+ int healing	เป็นค่าที่บอกว่า potion นี้สามารถเพิ่มเลือดได้กี่หน่วย
---------------	--

4.4.2 Constructor

+ HealPotion()	เรียก constructor ของ Item โดยใส่ชื่อของ Potion เข้าไปและ Set ค่า URL กับ healing โดยมีค่าเริ่มต้นที่ 50
----------------	--

4.4.3 Methods

+ void heal(Unit e)	เพิ่มเลือดให้ Unit ที่ถูกเลือกโดยเลือกที่ถูกเพิ่มแล้วจะมีค่ามากที่สุดไม่เกิน maxHealth และหลังจากเพิ่มเลือดเสร็จจะลดปริมาณของ potion นี้ลง
---------------------	--

4.5 Class RevivePotion = ขวดยาชุบชีวิต ซึ่ง extends มาจาก Items

4.5.1 Constructor

+ RevivePotion()	เรียก constructor ของ Item โดยใส่ชื่อของ Potion เข้าไปและ Set ค่า URL
------------------	---

4.5.2 Methods

+ void revive(Unit e)	ทำการชุบชีวิตของตัวละครที่ตายแล้วโดยจะ reset เลือดให้แต่จะไม่รีเซ็ตโล่ให้ และหลังจาก
-----------------------	--

	ซุบชีวิตแล้วจะทำการลดปริมาณของ potion ชนิดนี้ลง
--	---

5.Package entity.unit

เป็น package ที่เก็บ class ของ unit ทั้งหมดที่มีในเกมนี้ โดยทุกๆ class จะ extends มาจาก class Unit จาก entity.base

5.1 Class Buff = Unit ที่สามารถเพิ่มพลังโจมตีและเพิ่มโลให้ Unit อื่นได้

5.1.1 Field

- int BuffShield	ค่าความสามารถในการเพิ่มโลให้ Unit อื่น โดยมีค่าเริ่มต้นเป็น 20
- int BuffAttack	ค่าความสามารถในการเพิ่มพลังโจมตีให้ Unit อื่น โดยมีค่าเริ่มต้นเป็น 30

5.1.2 Constructor

+ Buff(String name,String URL)	เรียก Constructor ของ Unit และ set ค่าชื่อ และ URL ของตัวละคร
--------------------------------	---

5.1.3 Methods

+ int getBuffAttack()	getter ของ BuffAttack
+ void setBuffAttack(int buffAttack)	Setter ของ BuffAttack โดยค่า BuffAttack ต้องมากกว่า 0
+ int getBuffShield()	getter ของ BuffShield
+ void setBuffShield(int buffShield)	Setter ของ BuffShield โดยค่า BuffShield ต้องมากกว่า 0

+ void reset()	Reset ค่าเลือดและโล่ของตัวละครเพื่อให้พร้อมเล่นตาใหม่
+ void levelUp()	เพิ่มค่า stat ต่างๆของตัวละครเมื่อถูก uplevel และเรียก reset() เพื่อปรับให้เลือดกับโล่เต็มเหมือนเดิม
+ void buffATK(Unit e)	เพิ่มพลังโจมตีของ Unit ที่รับมาเท่ากับค่าของ BuffAttack
+ void buffShield(Unit e)	เพิ่มค่าโล่ของ Unit ที่รับมาเท่ากับค่าของ BuffShield (ถ้าUnitที่รับมาเป็นTankจะเพิ่ม magic shield ให้ด้วย)

5.2 Class DPS = Unit ที่มีพลังโจมตีเป็นจุดเด่นแต่จะมีเลือดน้อย

5.2.1 Constructor

+ DPS(String name,String URL)	เรียก Constructor ของ Unit และ set ค่าชื่อและ URL ของตัวละคร และปรับค่าพลังโจมตีให้สูงกว่า Unit ปกติ
-------------------------------	--

5.2.2 Methods

+ void levelUp()	เพิ่มค่า stat ต่างๆของตัวละครเมื่อถูก uplevel และเรียก reset() เพื่อปรับให้เลือดกับโล่เต็มเหมือนเดิม
+ void reset()	Reset ค่าเลือดและโล่ของตัวละครเพื่อให้พร้อมเล่นตาใหม่

5.3 Class Heal = Unit ที่สามารถเพิ่มเลือดให้ Unit อื่นๆในทีมได้

5.3.1 Field

- int heal	เก็บค่าที่บอกว่าสามารถเพิ่มเลือดให้ Unit อื่นได้กี่หน่วย
------------	--

5.3.2 Constructor

+ Heal(String name,String URL)	เรียก Constructor ของ Unit และ set ค่าชื่อและ URL ของตัวละคร และมีการ set ค่าเริ่มต้นของ heal
--------------------------------	---

5.3.3 Methods

+ int getHeal()	Getter ของ Field Heal
+ void setHeal(int heal)	Setter ของ Field Heal โดยค่า Heal จะต้องมากกว่ากับ 0
+ void Heal(Unit e)	เพิ่มเลือดให้กับ Unit ที่รับเข้ามาโดยเลือดหลังจากถูกเพิ่มจะต้องไม่มากกว่า maxHealth ของตัวละครนั้น
+ void levelUp()	เพิ่มค่า stat ต่างๆของตัวละครเมื่อถูก uplevel และเรียก reset() เพื่อปรับให้เลือดกับโล่เต็มเหมือนเดิม
+ void reset()	Reset ค่าเลือดและโล่ของตัวละครเพื่อให้พร้อมเล่นตาใหม่

5.4 Class Mage = Unit ที่มีพลังโจมตีเป็นเวทย์ซึ่งจะสามารถโจมตี Unit อื่นที่ไม่ใช่ Tank แบบไม่ต้องคำนวณพลังโจมตีลดเกราะ

5.4.1 Field

+ int spellDamage	เก็บค่าพลังโจมตีแบบเวทย์ของ Mage โดยมีค่าเริ่มต้นเป็น 100
-------------------	---

5.4.2 Constructor

+ Mage(String name,String URL)	เรียก Constructor ของ Unit และ set ค่าชื่อและ URL ของตัวละคร
--------------------------------	--

5.4.3 Methods

+ int getSpellDamage()	Getter ของ Field spellDamage
+ void setSpellDamage(int spellDamage)	Setter ของ Field spellDamage
+ void levelUp()	เพิ่มค่า stat ต่างๆของตัวละครเมื่อถูก uplevel และเรียก reset() เพื่อปรับให้เลือดกับโล่เต็มเหมือนเดิม
+ void reset()	Reset ค่าเลือดและโล่ของตัวละครเพื่อให้พร้อมเล่นตาใหม่

5.5 Class Tank = Unit ที่มีเลือดมากกว่า Unit อื่นๆและมีเกราะเวทย์เพื่อป้องกันการโจมตีของ Mage ได้

2.1.1 Field

# int maxShieldMagic	ค่ามากที่สุดของเกราะเวทย์ โดยมีค่าเริ่มต้นเป็น150
# int shieldMagic	ค่าเกราะเวทย์เพื่อหักลบการโจมตีของ mage โดยมีค่าเริ่มต้นเป็น150

2.1.2 Constructor

+ Tank(String name,String URL)	เรียก Constructor ของ Unit และ set ค่าชื่อและ URL ของตัวละคร
--------------------------------	--

2.1.3 Methods

+ int getMaxShieldMagic()	Getter ของ maxShieldMagic
+ void setMaxShieldMagic(int maxShieldMagic)	Setter ของ maxShieldMagic
+ int getShieldMagic()	Getter ของ shieldMagic
+ void setShieldMagic(int shieldMagic)	Setter ของ shieldMagic
+ int decreaseShieldMagic(int damage)	ลดเกราะเวทย์ของ Tank ตาม spellDamage ที่ได้รับมา และคืนค่า Damage ที่เหลือจากการหักลบเพื่อไปลบต่อที่เลือด
+ void levelUp()	เพิ่มค่า stat ต่างๆของตัวละครเมื่อถูก uplevel และเรียก reset() เพื่อปรับให้เลือดกับโล่เต็มเหมือนเดิม
+ void reset()	Reset ค่าเลือดและโล่ของตัวละครเพื่อให้พร้อมเล่นตาใหม่

6.Package logic

เป็น package ที่เก็บ class ของ logic ทั้งหมดที่ใช้ในการเล่นเกมนี้ รวมถึงการใช้ควบคุมการเล่นเกมทั้งหมดด้วย

6.1 Class Player = เป็น class สำหรับการจัดการ Player

6.1.1 Field

+ int money	เก็บค่าเงินของผู้เล่น
+ ArrayList<Unit> unitsDeck	เก็บค่า Unit ทั้งหมดของผู้เล่น
+ ArrayList<Items> itemsDeck	เก็บค่า Items ทั้งหมดของผู้เล่น

6.1.2 Constructor

+ Player()	-ตั้งค่าเงินของผู้เล่น เริ่มต้นที่ 100 -ตั้งค่า Unit ที่ผู้เล่นมีโดยเริ่มต้นผู้เล่นจะมี Unit ทุกตัวที่ level 1 -ตั้งค่า Items ที่ผู้เล่นมีโดยตอนเริ่มต้นผู้เล่นจะไม่มี Item เลย
------------	---

6.1.3 Methods

+ void levelUp_all()	เพิ่ม level ให้ Unit ทุกตัวที่ player มี 1 level
+ void levelUp_all(int level)	เพิ่ม level ให้ Unit ทุกตัวที่ player มีตามจำนวน level ที่รับมา
+ int getMoney()	Getter ของ field money
+ void SetMoney(int money)	Setter ของ field money โดยจำนวนเงินที่มีจะต้องมากกว่าเท่ากับ 0
+ ArrayList<Unit> getUnitsDeck()	Getter ของ field unitsDeck

+ void setUnitsDeck(ArrayList<Unit> unitsDeck)	Setter ของ field unitsDeck
+ ArrayList<Items> getItemsDeck()	Getter ของ field ItemsDeck
+ void setItemsDeck(ArrayList<Items> itemsDeck)	Setter ของ field itemsDeck
+ Boolean isAllDead()	จะคืนค่า จริง เมื่อตัวละครทุกตัวใน unitsDeck ตายหมดแล้ว นอกนั้นจะคืนค่า เท็จ
+ void pay(int cost)	หักเงินของผู้เล่นตาม cost
+ void reward(int money)	เพิ่มเงินผู้เล่นตาม money

6.2 Class Enemy = เป็น class สำหรับการจัดการ Enemy

6.2.1 Field

+ ArrayList<Unit> unitsDeck	เก็บค่า Unit ทั้งหมดของผู้เล่น
-----------------------------	--------------------------------

6.2.2 Constructor

+ Enemy()	-ตั้งค่า Unit ที่ enemy มีโดยเริ่มต้นจะมี Unit ทุกตัวที่ level 1
-----------	--

6.2.3 Methods

+ void levelUp_all()	เพิ่ม level ให้ Unit ทุกตัวที่ enemy มี 1 level
+ void levelUp_all(int level)	เพิ่ม level ให้ Unit ทุกตัวที่ enemy มีตาม จำนวน level ที่รับมา
+ ArrayList<Unit> getUnitsDeck()	Getter ของ field unitsDeck
+ void setUnitsDeck(ArrayList<Unit> unitsDeck)	Setter ของ field unitsDeck

+ Boolean isAllDead()	จะคืนค่า จริง เมื่อตัวละครทุกตัวใน unitsDeck ตายหมดแล้ว นอกนั้นจะคืนค่า เท็จ
-----------------------	--

6.3 Class GameLogic = เป็น class รวบรวม logic ทั้งหมดที่ใช้ในการเล่นเกมนี

6.3.1 Field

- static Player player	เก็บ class player
- static ArrayList<Unit> playerField	เก็บตัวละครที่ player เลือกให้ลงสนาม
- static Enemy enemy	เก็บ class enemy
- static ArrayList<Unit> enemyField	เก็บตัวละครที่ enemy เลือกให้ลงสนาม
- static boolean isGameEnd	เก็บ boolean เพื่อเช็คว่างेमจบหรือยัง
- static boolean isPlayerWin	เก็บ boolean เพื่อเช็คว่างู่เล่นชนะหรือยัง
- static boolean isPlayerLose	เก็บ boolean เพื่อเช็คว่างู่เล่นแพ้หรือยัง
- static int level	เก็บ level ของเนว่าตอนนี้ผู้เล่นอยู่ด่านไหนแล้ว

6.3.2 Methods

+ static void gameStart()	-สร้าง player -สร้าง enemy ที่ level 2
+ static void resetEnemy()	-สร้าง enemy ที่ level 2 -reset level ให้เหลือ 1 (เริ่มเกมใหม่)
+ static void createEnemy(int level)	สร้าง enemy ตาม level ที่ต้องการและสุ่มตัวมาลงใน enemyField
+ static void reset_gameEnd()	-reset ค่า Boolean ให้เป็น false ทั้งหมดหลังจบเกม

+ static void updatePlayerField(ArrayList<Unit> playerField)	รับ playerField มาจากหน้า gui ที่ผู้เล่นเลือก ตัวละครมาตั้งค่าใน GameLogic
+ static void enemyAttack()	คำนวณ Damage ที่ enemy โจมตีเรา พร้อมทั้งส่งตัวที่โจมตีและถูกโจมตีให้ด้วย (Unit Tank จะถูกโจมตีก่อนเสมอ)
+ static void calculatePlayerAttack(int attacker,int defender)	คำนวณ damage ที่ผู้เล่นโจมตี enemy โดย เราจะรับ index ของ attacker กับ defender มาจากหน้า gui ที่ผู้เล่นคลิกโจมตี
+ static void healAll(int indexHealer)	เพิ่มเลือดให้ Unit ทุกตัวใน playerField ตาม ความสามารถของ healer ใน indexhealer ของ playerField
+ static void buffAll(int bufferIndex)	เพิ่มโล่และพลังโจมตีให้ Unit ทุกตัวใน playerField ตามความสามารถของ buffer ใน bufferIndex ของ playerField
+ static void usePotion(int indexPotion, int indexUnit)	ใช้ความสามารถของ Potion กับ Unit ที่ถูก เรียกโดยจะรับ index ของทั้ง potion และ Unit มาจากหน้า gui
+ static void updateGameStatus()	Update status ของ game หลังจากการ โจมตีทุกครั้งว่าเกมจบหรือยัง โดยจะเรียก checkPlayerWin() กับ checkPlayerLose()
+ static boolean checkPlayerLose()	คืนค่า จริงก็ต่อเมื่อทุก Unit ใน playerField ตายหมดแล้ว ที่เหลือเป็นเท็จ
+ static boolean CheckPlayerwin()	คืนค่า จริงก็ต่อเมื่อทุก Unit ใน enemyField ตายหมดแล้ว ที่เหลือเป็นเท็จ
+ static void checkGameEnd()	หลังจากการโจมตีจะเช็คว่ามีคนแพ้หรือชนะ หรือยังโดยจะเรียก updateGameStatus()

	ซึ่งถ้ารู้ผลแพ้ชนะแล้ว isGameEnd จะถูกเปลี่ยนเป็น true นอกนั้นจะเป็นเท็จ
+ static void resetPlayerUnit()	Reset ทุก Unit ของ player ให้กลับมามีเลือดและโล่เต็มเหมือนเดิม และปรับพลังโจมตีให้กลับมาเหมือนเดิมด้วยการเรียก reset() ของแต่ละ Unit
+ static Player getPlayer()	Getter ของ Field player
+ static void setPlayer(Player player)	Setter ของ Field player
+ static ArrayList<Unit> getPlayerField()	Getter ของ Field playerField
+ static void setPlayerField(ArrayList<Unit> playerField)	Setter ของ Field playerField
+ static Enemy getEnemy()	Getter ของ Field enemy
+ static void setEnemy(Enemy enemy)	Setter ของ Field enemy
+ static ArrayList<Unit> getEnemyField()	Getter ของ Field enemyField
+ static void setEnemyField(ArrayList<Unit> enemyField)	Setter ของ Field enemyField
+ static boolean isGameEnd()	Getter ของ Field isGameEnd
+ static void setGameEnd(boolean isGameEnd)	Setter ของ Field isGameEnd
+ static int getLevel()	Getter ของ Field level
+ static void setLevel(int level)	Setter ของ Field level
+ static boolean isPlayerWin()	Getter ของ Field isPlayerWin
+ static void setPlayerWin(boolean isPlayerWin)	Setter ของ Field isPlayerWin
+ static boolean isPlayerLose()	Getter ของ Field isPlayerLose

+ static void setPlayerLose(boolean isPlayerLose)	Setter ของ Field isPlayerLose
---	-------------------------------

7.Package stage

เป็น package ที่เก็บ class และ fxml ทั้งหมดที่ใช้ในการแสดงผล gui

7.1 Class Menu

2.1.1 Field

- Scene menuScene	เก็บ scene ของหน้า menu เริ่มต้นไว้
-------------------	-------------------------------------

2.1.2 Constructor

+ Menu()	สั่ง startgame จาก GameLogic สร้าง scene ของ menu จากไฟล์ fxml
----------	--

2.1.3 Methods

+ Scene getMenuScene()	Getter field ของ menuScene
+ void setMenuScene(Scene menuScene)	Setter field ของ menuScene