

Knight Adventure



คณะผู้จัดทำ

นายธีรศักดิ์	��ะตันต๋อง	620510645
นายเพชร	สุขสำราญ	620510650
นางสาวพรสรวง	รังควร	620510651
นายภูตะวัน	สาระสุข	620510658

อาจารย์ที่ปรึกษา

ดร.ประภาพร เฑชอังกูร

ผศ.ดร.เมทินี เขียวกันยะ

โครงงานนี้เป็นส่วนหนึ่งของกระบวนการวิชา204211การเขียนโปรแกรมเชิงวัตถุ

ภาควิชาวิทยาการคอมพิวเตอร์ คณะวิทยาศาสตร์

มหาวิทยาลัยเชียงใหม่

สารบัญ

หน้า

บทนำ	1
Problem and Problem Analysis	2
Class Design	3
Class Relationships	12
Tool and Technique	13
Screen Capture	14
Program Installation	18
Reference	22

บทนำ

ชื่อกลุ่ม : Sleepy Wake Up

หัวข้อโครงงาน : Knight Adventure

ผู้จัดทำโครงงาน :

นายธีรศักดิ์	ทิศันต๊อง	620510645	sec 002
นายเพชร	สุขสำราญ	620510650	sec 002
นางสาวพรสรวง	รังควร	620510651	sec 002
นายภูตะวัน	สาระสุข	620510658	sec 002

หน้าที่ในกลุ่ม

Coder : นายธีรศักดิ์ ทิศันต๊อง , นายเพชร สุขสำราญ

Graphic Designer : นายภูตะวัน สาระสุข

Documentation : นางสาวพรสรวง รังควร

Presentation : นายธีรศักดิ์ ทิศันต๊อง , นายเพชร สุขสำราญ ,
นายภูตะวัน สาระสุข , นางสาวพรสรวง รังควร

Problem and Problem Analysis

โจทย์ปัญหา

เนื่องจากผู้จัดทำต้องการสร้างโปรแกรมให้สอดคล้องกับวิชาการเขียนโปรแกรมเชิงวัตถุ(Object-Oriented Programming) ซึ่งเราจึงอยากจัดทำเกมส์ที่สามารถนำมาใช้งานได้จริง และสามารถนำไปต่อยอดได้ในอนาคตและนำความรู้ที่ได้จากการค้นคว้าไปฝึกทักษะการเขียนโปรแกรมต่อไป

การวิเคราะห์โจทย์

- การออกแบบตัวเกมและคิระบบของเกมอย่างไรให้มีความง่าย ไม่ซับซ้อน ต่อผู้เล่น
- พัฒนาตัวเกมด้วยภาษา Java และหลักการเขียนโปรแกรมเชิงวัตถุอย่างไรให้มีประสิทธิภาพ
- การออกแบบ Graphic ภายในเกมยังไ้ให้มีความสอดคล้องกับตัวเกม

ขั้นตอนการแก้ไขปัญา

- ✚ ออกแบบโปรแกรมเพื่อสร้างเป็นเชิงวัตถุ ให้สามารถทำงานอย่างเป็นระบบ
- ✚ ศึกษาการสร้างโครงสร้างของหน้าต่างเกม การใส่รูปต่างๆและไฟล์เสียง
- ✚ ออกแบบภาพจากพื้นหลังต่างๆในเกม
- ✚ สร้างบล็อกในรูปแบบต่างๆ ภาพพื้นหลัง รวมถึงการใส่เสียงเพลงประกอบ
- ✚ ทดสอบการทำงานของโปรแกรม แก้ไข ปรับปรุง ข้อผิดพลาดที่เกิดขึ้น

Class design

Package GameScreen:

Game

```
+end game: Boolean
+play: Boolean
+htp: HowToPlay
-SG : SaveGame
-frame : JFrame
+ Game(Boolean)
+ setGame(): void
+ loadGame(): void
+ changePlayState(): void
+ gameOver(): void
+ VictoryGame(): void
+ resetGame(): void
```

GameScreen

```
+ player : Player
+ GameScreen()
+ resetGameScreen(): void
```

Score

```
- frame: JFrame
- score : int
+ ShowScore: JLabel
+ Score()
+setScore(): void
+ resetScore(): void
+ addScore(): void
+ showScore(): void
+ getScore(): int
+ destroyScore(): void
```

menuQuitGame

```
- frame: JFrame
- yesButtonBasic: ImageIcon
- noButtonBasic: ImageIcon
- noButtonEntered : ImageIcon
- quitLabel : JLabel
- yesButton: JButton
- noButton: JButton
-es: EffectSound
+ menuQuitGame()
- showMenuQuitGame(): void
- dontShowMenuQuitGame(): void
+ menuQuitGame(): void
```


Stage

- + Background: JLabel
- frame: JFrame
- tabNextStage: JLabel
- SG: SaveGame
- + stage: int
- + level: int
- difficulty: String[]
- leveldifficulty: JLabel
- TextStage: JLabel

- +Stage()
- +getStage(): int
- +setStage(): void
- + setBackground (): void
- + passStage(): void
- setTextLevel(): void
- nextLevel(): void
- + resetStage(): void

MenuGameOver

- frame: JFrame
- playButtonBasic: ImageIcon
- playButtonEntered: ImageIcon
- QuitButtonBasic: ImageIcon
- QuitButtonEntered: ImageIcon
- playButton: JButton
- QuitButton: JButton
- highS: HighScore
- ShowHighScore: JLabel
- ShowLastScore: JLabel
- es: EffectSound

- + MENUGameOver()
- +showGameOver(): void
- dontShowGameOver(): void
- callHighScore(): void
- setLastScore(): void

coin

- iconCoin : ImageIcon
- + COIN : JLabel
- numCoin: int
- +ShowCoin : JLabel
- + frame: JFrame

- +Coin()
- + addCoin(): void
- + subCoin(): void
- + reduceCoin(): void
- + showCoin(): void
- + resetCoin(): void

HowToPlay

- HowToPlay: JLabel
- exitShop: JButton
- es: EffectSound
- frame: JFrame

- + HowToPlay()
- + showHowToPlay(): void
- + dontShowHowToPlay(): void

menuVictoryGame

- frame: JFrame
- playButtonBasic: ImageIcon
- playButtonEntered :
ImageIcon
- QuitButtonBasic: ImageIcon
- victoryGameLabel: JLabel
- playButton: JButton
- QuitButton: JButton
- highs: HighScore
- highScore : int
- ShowHighScore: JLabel
- ShowLastScore: JLabel
- es : EffectSound

- + menuVictoryGame()
- + showVictoryGame(): void
- dontShowVictoryGame(): void
- callHighScore() : void
- setLastScore(): void

Heart

- iconHeart: ImageIcon
- numHeart: int
- maxHeart : int
- heart : JLabel
- frame: JFrame

- + Heart()
- + setHeart(): void
- + getNumHeart(): int
- + getMaxHeart(): int
- + increaseHeart(): void
- + decreaseHeart(): void
- + increaseIconHeart(): void
- + decreaseIconHeart(): void
- + resetHealth(): void

Character

- lv_armor: int
- playerAction : char
- playerU: ImageIcon
- playerD: ImageIcon
- playerL: ImageIcon
- playerR: ImageIcon
- + characterPlayer: JLabel
- frame: JFrame

- + Character()
- + changeImageAction(): void
- + changeImagePlayer(): void
- + changeImageActionToUP(): void
- + changeImageActionToDown(): void
- + changeImageActionToLeft(): void
- + changeImageActionToRight(): void
- + resetCharacter(): void

Armor

- iconArmor: ImageIcon
- numArmor: int
- maxArmor: int
- armor: JLabel
- frame: JFrame

- + Armor()
- + setHeart(): void
- + getNumArmor(): int
- + getMaxArmor(): int
- + increaseArmor(): void
- + decreaseArmor(): void
- + increaseIconArmor(): void
- + decreaseIconArmor(): void
- + resetArmor(): void

Package Map:

Map

```
#thread : Thread
#numMap : int
#xObstacle : XAxisObstacle
#NoPointShot : int
#shotSequenceRow1 : ArrayList<Integer>
#shotSequenceRow2 : ArrayList<Integer>
#shotSequenceRow3 : ArrayList<Integer>
#shotSequenceRow4 : ArrayList<Integer>
#shotSequenceRow5 : ArrayList<Integer>
#shotSequenceRow6 : ArrayList<Integer>
#shotSequenceRow7 : ArrayList<Integer>
#shotSequenceRow8 : ArrayList<Integer>
#show : Boolean
+time : int
-maxtime : int
-Cooldown : int
```

```
+Map
+run() : void
-calShow() : void
-nextObstacles() : void
-runObstacles() : void
#createXObstacle() : void
#setObstacle() : void
#setRow1() : void
#setRow2() : void
#setRow3() : void
#setRow4() : void
#setRow5() : void
#setRow6() : void
#setRow7() : void
#setRow8() : void
+setDefaultMap() : void
+increasingDifficulty() : void
+dontShowObstacles() : void
-operateObstacles() : boolean
- operateObstacles(int) : boolean
+setMap() : void
```

map1-10

```
+map1-10(Stage)
#createXObstacle() : void
#setObstacles() : void
#setRow1() : void
#setRow2() : void
#setRow3() : void
#setRow4() : void
#setRow5() : void
#setRow6() : void
#setRow7() : void
#setRow8() : void
```

map2

```
+map2(Stage)
#createXObstacle() : void
#setObstacles() : void
#setRow1() : void
#setRow2() : void
#setRow3() : void
#setRow4() : void
#setRow5() : void
#setRow6() : void
#setRow7() : void
#setRow8() : void
```

map3

```
+map3(Stage)
#createXObstacle() : void
#setObstacles() : void
#setRow1() : void
#setRow2() : void
#setRow3() : void
#setRow4() : void
#setRow5() : void
#setRow6() : void
#setRow7() : void
#setRow8() : void
```


map4

```
+map4(Stage)
#createXObstacle() : void
#setObstacles() : void
#setRow1() : void
#setRow2() : void
#setRow3() : void
#setRow4() : void
#setRow5() : void
#setRow6() : void
#setRow7() : void
#setRow8() : void
```

map5

```
+map5(Stage)
#createXObstacle() : void
#setObstacles() : void
#setRow1() : void
#setRow2() : void
#setRow3() : void
#setRow4() : void
#setRow5() : void
#setRow6() : void
#setRow7() : void
#setRow8() : void
```

map6

```
+map6(Stage)
#createXObstacle() : void
#setObstacles() : void
#setRow1() : void
#setRow2() : void
#setRow3() : void
#setRow4() : void
#setRow5() : void
#setRow6() : void
#setRow7() : void
#setRow8() : void
```

map7

```
+map7(Stage)
#createXObstacle() : void
#setObstacles() : void
#setRow1() : void
#setRow2() : void
#setRow3() : void
#setRow4() : void
#setRow5() : void
#setRow6() : void
#setRow7() : void
#setRow8() : void
```

map8

```
+map8(Stage)
#createXObstacle() : void
#setObstacles() : void
#setRow1() : void
#setRow2() : void
#setRow3() : void
#setRow4() : void
#setRow5() : void
#setRow6() : void
#setRow7() : void
#setRow8() : void
```

map9

```
+map9(Stage)
#createXObstacle() : void
#setObstacles() : void
#setRow1() : void
#setRow2() : void
#setRow3() : void
#setRow4() : void
#setRow5() : void
#setRow6() : void
#setRow7() : void
#setRow8() : void
```

map10

```

+map10(Stage)
#createXObstacle() : void
#setObstacles() : void
#setRow1() : void
#setRow2() : void
#setRow3() : void
#setRow4() : void
#setRow5() : void
#setRow6() : void
#setRow7() : void
#setRow8() : void

```

Package Shop:

Item

```

#canSell : ImageIcon
#cannotSell : ImageIcon
#price : int
#item : JLabel
#frame : JFrame
#es : EffectSound

```

```

+Item()
+sellItem() : void
+canSellItem() : void
+purchaseStatus() : void
+showItem() : void
+dontShowItem() : void

```

Shop

```

-thread : Thread
-frame : JFrame
+SHOWSHOP : Boolean
-shop : JLabel
-exitShop : JButton
-coin : JLabel
-isExitShop : Boolean
-es : EffectSound
-num : JLabel

```

```

+Shop
+run() : void
-isShowShop() : Boolean
-showShop() : void
+dontShowShop() : void

```

ItemArmor

```

-LvArmor : int

```

```

+ItemArmor(int,int,int,int)
+SellItem() : void
+canSellItem() : boolean

```

ItemHeart

```

+ItemHeart(int,int,int,)
+SellItem() : void
+canSellItem() : boolean

```

Package KnightAdventure:

Sound

- clip: Clip
- + Sound()
- + playmusic(): void
- + Play(): void
- + Stop(): void
- + getClip(): Clip

Window

- +frame: JFrame
- +Window()

GameMenuScreen

- BackGround: JLabel
- continueF: ImageIcon
- continueT: ImageIcon
- newGameT: ImageIcon
- quit: ImageIcon
- continueB : JButton
- newgameB: JButton
- tapContinue: JButton
- tabnewgame : JButton
- tabquit: JButton
- savegame: Boolean
- frame: JFrame
- +GameMenuScreen()
- exitMainMenu(): void

SaveGame

- line: int
- n : int
- list: int[]
- dirPath: String
- + SaveGame()
- + load(): void
- + save(): void
- + save(int, int, int, int): void
- + getSave(): int
- + getNumHeart(): int
- + getNumArmor(): int
- + getScore(): int
- + getCoin(): int
- + getStage(): int

KeyInput

- A: boolean
- N: Boolean
- P: Boolean
- L: Boolean
- + KeyInput(Game)
- + keyPressed(): void
- + keyRelease(): void

HighScore

- HighScore: int
- s: String
- dirpath: String
- + HighScore()
- + setHighScore(int): void
- + getHighScore(): int

EffectSound

- + EffectSound()
- + playEffect(String): void

KnightAdventure

- + SCREEN WIDTH: int
- + SCREEN HIGHT: int
- + main(String): void

Package Object:

Player

```
-x : int
-y : int
-PLAYER : JLabel

+Player(Game,GameScreen)
+setPlayer(int,int,int) : void
+getX() : int
+getY() : int
+isDied() : Boolean
+damaged(int) : void
+addArmor() : void
+receiveHeart() : void
+loseHeart() : void
+playerMoveUp() : void
+ playerMoveDown() : void
+ playerMoveLeft() : void
+ playerMoveRight() : void
+resetLocationPlayer() : void
+soundwalk() : void
+resetPlayer() : void
```

LXAxisObstacle

```
+LXAxisObstacle()
+isEndOfShooting() : Boolean
+move() : void
+resetLocation() : void
+setDefaultXObstacle() : void
#setIconObstacle() : void
```

LArrow

```
+LArrow(int)
#setIconObstacle() : void
```

LFireball

```
+LFireball(int)
#setIconObstacle() : void
```

LMagicball

```
+LMagicball(int)
#setIconObstacle() : void
```

LSoulball

```
+LSoulball(int)
#setIconObstacle() : void
```

RFireball

```
+RFireball(int)
#setIconObstacle() : void
```

RMagicball

```
+RMagicball(int)
#setIconObstacle() : void
```

RSoulball

```
+RSoulball(int)
#setIconObstacle() : void
```

RArrow

```
+RArrow(int)
#setIconObstacle() : void
```

RXAxisObstacle

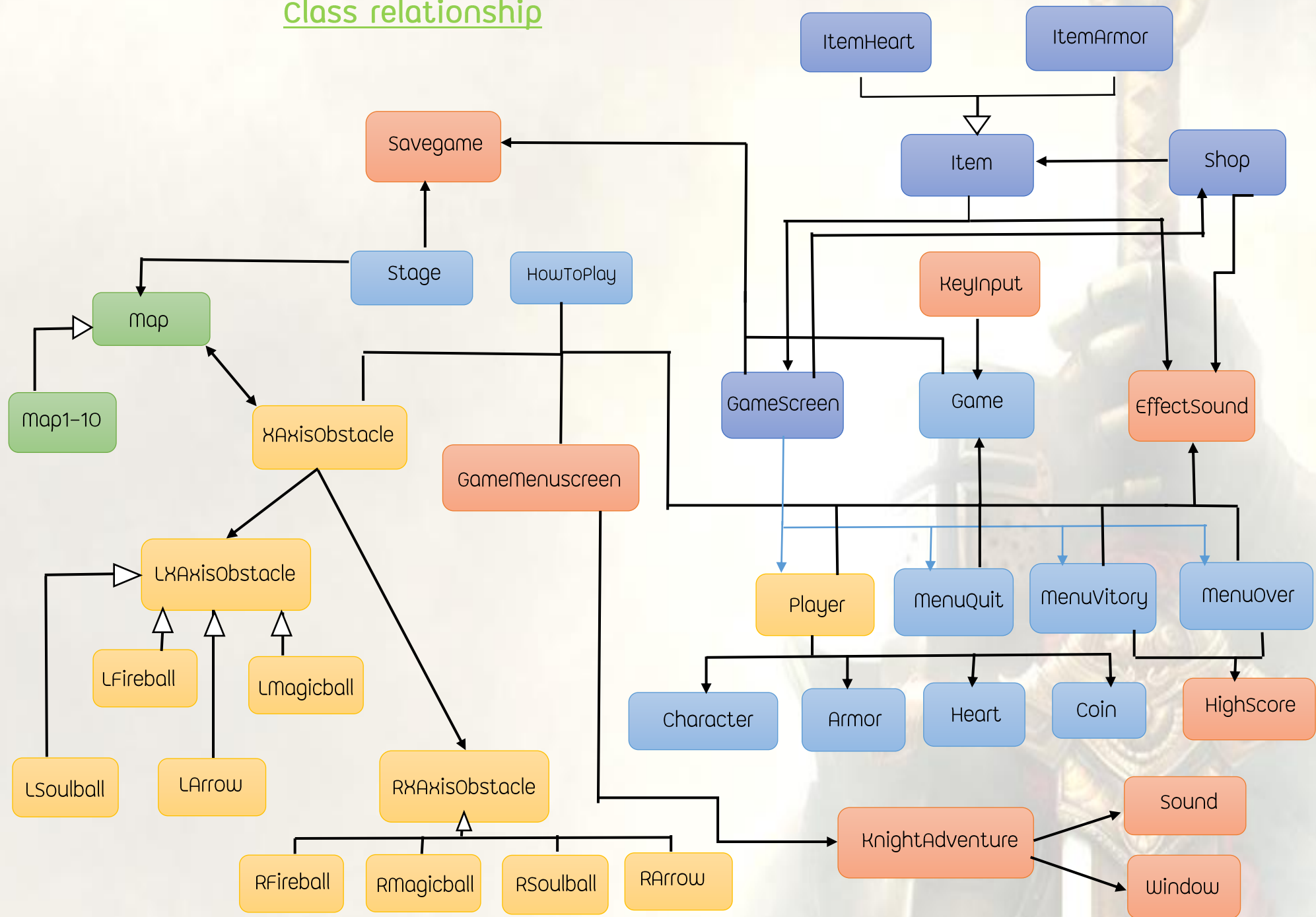
```
+RXAxisobstacle
+move() : void
+resetLocation() : void
+isEndOfShooting() : boolean
+setDefaultXobstacle() : void
+setIconObstacle() : void
```

XAxisObstacle

```
# XAxisObstacle: JLabel
# frame: JFrame
# damage: int
# speed: int
# x: int
# y: int
- shooting: boolean
# f_damage: int
# f_speed: int
# f_x: int
- f_shooting: Boolean
# sound: String

+ XAxisObstacle()
+ playSoundObstacle(): void
+ getX(): int
+ getY(): int
+ getDamage(): int
+ getShooting(): Boolean
+ getShooting(): void
+ setShooting(): void
+ setX(int): void
+ setLocationObstacle(): void
+ setSpeed(int): void
+ addDamage(): void
+ addSpeed(): void
+ addSpeed(int): void
+ hitShot(): void
+ endOfShooting(): void
+ showArrow(): void
+ dontShowArrow(): void
+ setDefaultXObstacle(): void
+ isEndOfShooting(): Boolean
+ move(): void
+ resetLocation(): void
+ isHitShot(): Boolean
```


Class relationship



Tool and Technique

Adobe Photoshop



ใช้ Adobe Photoshop ในการสร้างกราฟฟิกต่างๆภายในเกมส์

Apache NetBeans



ใช้โปรแกรม Apache NetBeans ในการสร้างระบบต่างๆภายในเกมส์

Launch4j



ใช้ Launch4j ในการทำตัวเกมส์

Adobe Premiere Pro



ใช้ Adobe Premiere Pro ในการตัดต่อเสียง

Screen Capture

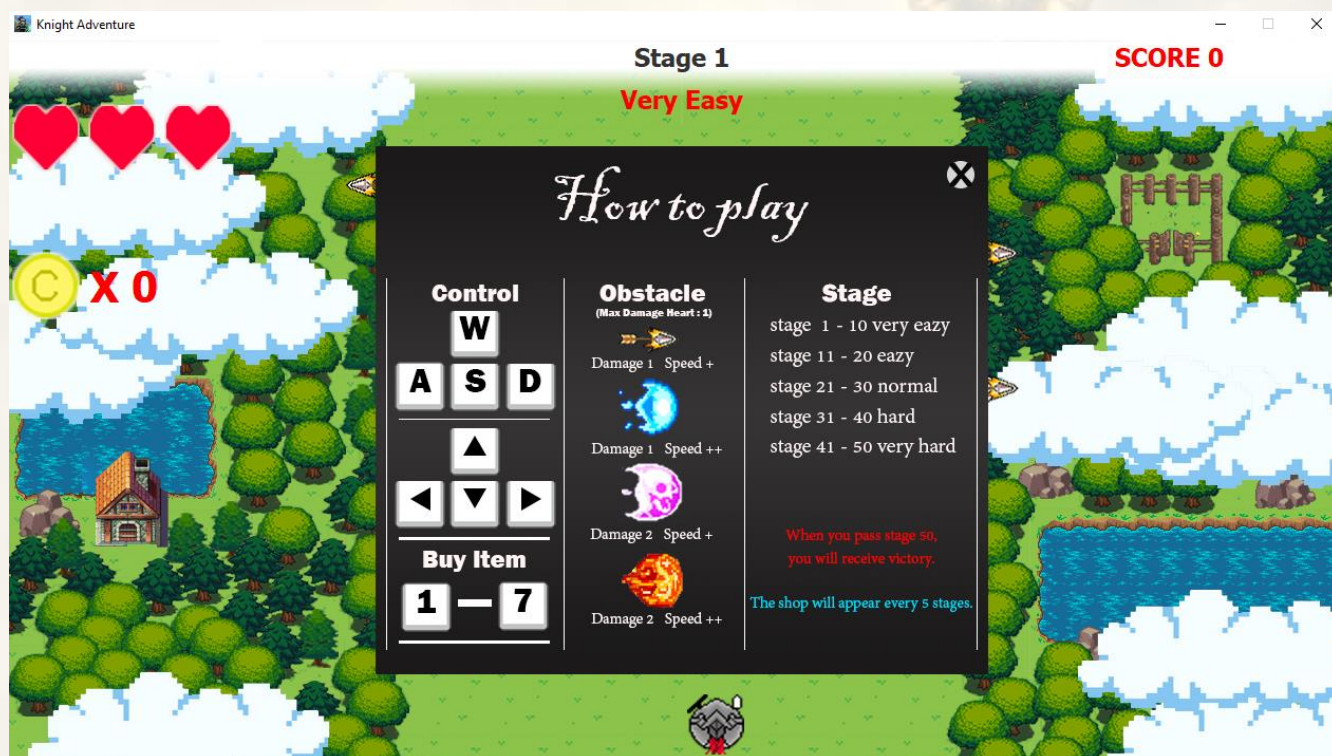
เมื่อเปิดตัวเกมขึ้นมาเราสามารถกดเริ่มเกมที่ new game หรืออยากเล่นต่อจากที่เล่นไว้ก่อนหน้านี้ให้กด Continue



เมื่อเริ่มเกมแล้วจะขึ้นหน้าต่างดังนี้



เกมจะบันทึกอวตารทุกครั้งที่ผ่านมา จะเริ่มบันทึกใหม่เมื่อตายหรือได้รับ ชัยชนะ ซึ่งเป้าหมายของเกมคือการผ่านด่านต่างๆ เพื่อไปให้ถึงเส้นชัยคือด่านสุดท้าย ซึ่งจะมีทั้งหมด 50 ด่าน เมื่อผ่านด่านที่ 50 ก็ชนะ ถ้าเราเผลอไปโดนอุปสรรค ก็จะทำการลดหัวใจลง 1 ดวง ถ้ามีเกราะก็จะลดเกราะก่อน ซึ่งในเกมจะมี interface ต่างๆ เช่น เหยี่ยว หัวใจ เกราะ คะแนน



การบังคับ

การบังคับตัวละครเราสามารถใช้ได้ทั้งสองแบบตามภาพข้างบนเลย ส่วนในการซื้อของในร้านค้าจะกดเลข 1-7 ตามภาพ

ศัตรู

ส่วนในค่านัศัตรูในเกม ตามภาพด้านบน จะมีอยู่ 4 ชนิดประกอบด้วย ธนู ลูกไฟ ลูกพลั้วเวท และลูกพลั้วธาตุมืด

- ธนู เป็นศัตรูที่ง่ายที่สุดเนื่องจากมี damage ที่จะลดเกราะแค่ 1 ชั้น
- ลูกบอลเวท จะยากขึ้นถัดจากธนู จะมีความไวที่เร็วขึ้น แต่จะมี damage ที่เท่ากับ ธนู
- ลูกบอลธาตุมืด จะมีความเร็วเป็นระดับเท่ากับธนู แต่จะมี damage ที่ลดเกราะ 2 ชั้น
- ลูกบอลไฟ จะมีความเร็วที่เพิ่มขึ้น และมี damage ลดเกราะ 2 ชั้น



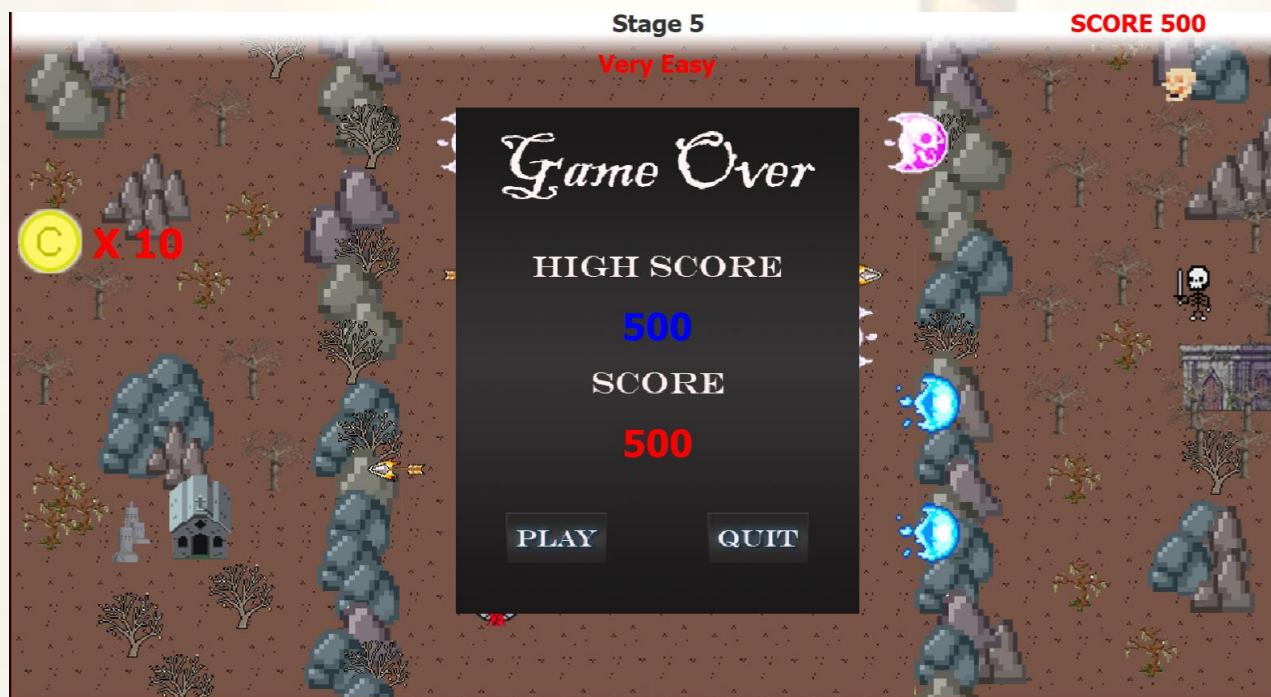
เหรียญ - จะได้รับก็ต่อเมื่อผ่านแต่ละด่าน และจะได้เพิ่มไปเรื่อยๆ เมื่อผ่านทุกๆ 5 ด่าน จะมีหน้าต่างร้านค้าขึ้นมาให้ซื้อหัวใจและเกราะ

หัวใจ - เป็นตัวกำหนดการจบเกม เมื่อเริ่มแรกมาเราจะมีหัวใจ 3 ดวง ถ้าเกิดว่าหัวใจหมดเกมก็จะจบทันทีที่สามารถซื้อได้จากร้านค้า เมื่อเราโค่นอุปสรรคโอมตี้ไม่ว่าศัตรูจะเป็นชนิดไหน หัวใจจะลดแค่ 1 ดวงและกลับไปเกิดใหม่ที่จุดเริ่มต้นของด่านนั้น

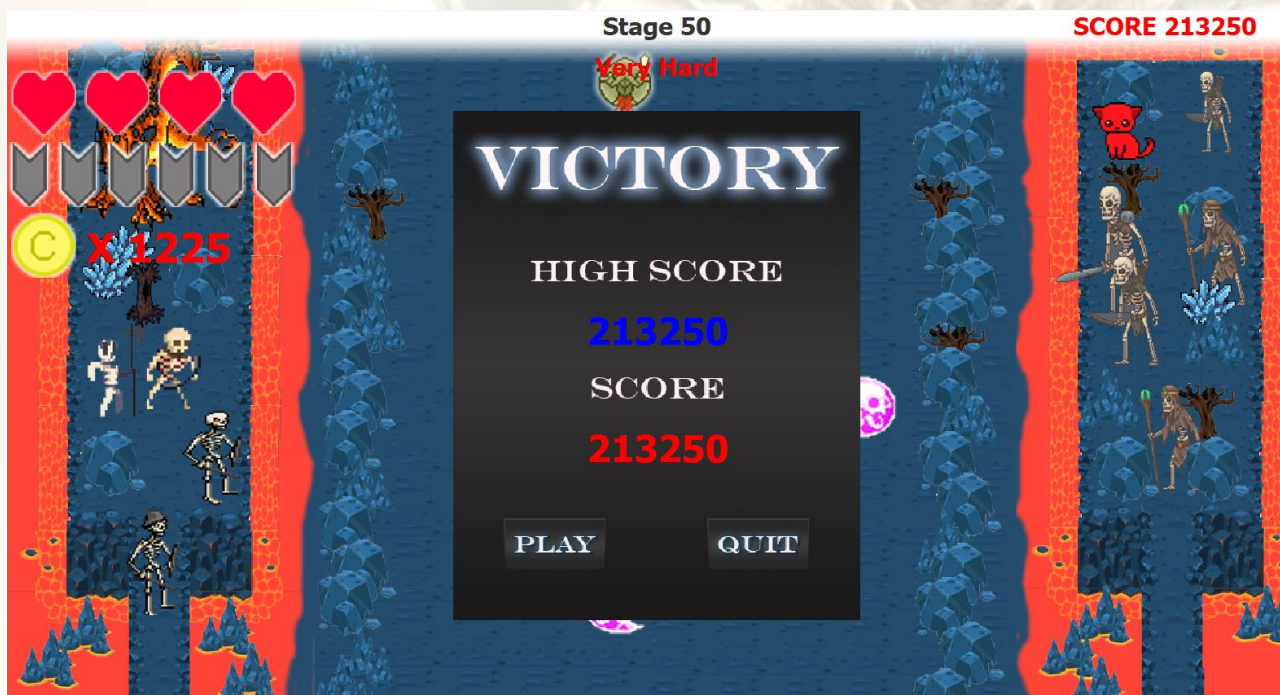
เกราะ - เป็นเกราะป้องกันหัวใจเมื่อเกราะหมดก็จะทำการลดหัวใจต่อไปสามารถซื้อได้จากร้านค้าได้เช่นกัน แต่เกราะจะแตกต่างจากหัวใจตรงที่ การลดของเกราะจะขึ้นกับ damage ของศัตรูชนิดนั้นๆ และเมื่อเราโค่นศัตรูโอมตี้เราจะสามารถเดินต่อไปได้ ตราบใดที่เรายังมีเกราะอยู่

คะแนน - เราจะได้รับคะแนนเมื่อผ่านแต่ละด่านเช่นเดียวกับเหรียญ ซึ่งคะแนนจะได้รับคือ $100 + (10 \times \text{จำนวนเหรียญที่เราีก่อนจะผ่านด่าน})$ นั่นก็คือถ้าเราใช้เหรียญน้อยเรายังจะได้คะแนนมากนั่นเอง

เมื่อหัวใจหมดเกมจะจบลงและเราสามารถเลือกเล่นใหม่หรือออกเกมได้ซึ่งหน้าต่างจะแสดงคะแนนที่ได้ และคะแนนสูงสุดที่ได้โดยทำไว้นั่นเอง



เช่นเดียวกับเมื่อเราผ่านด่านสุดท้ายก็คือด่านที่ 50 เราก็จะชนะเกมและสามารถเลือกที่จะกลับมาเล่นใหม่หรือออกเกม

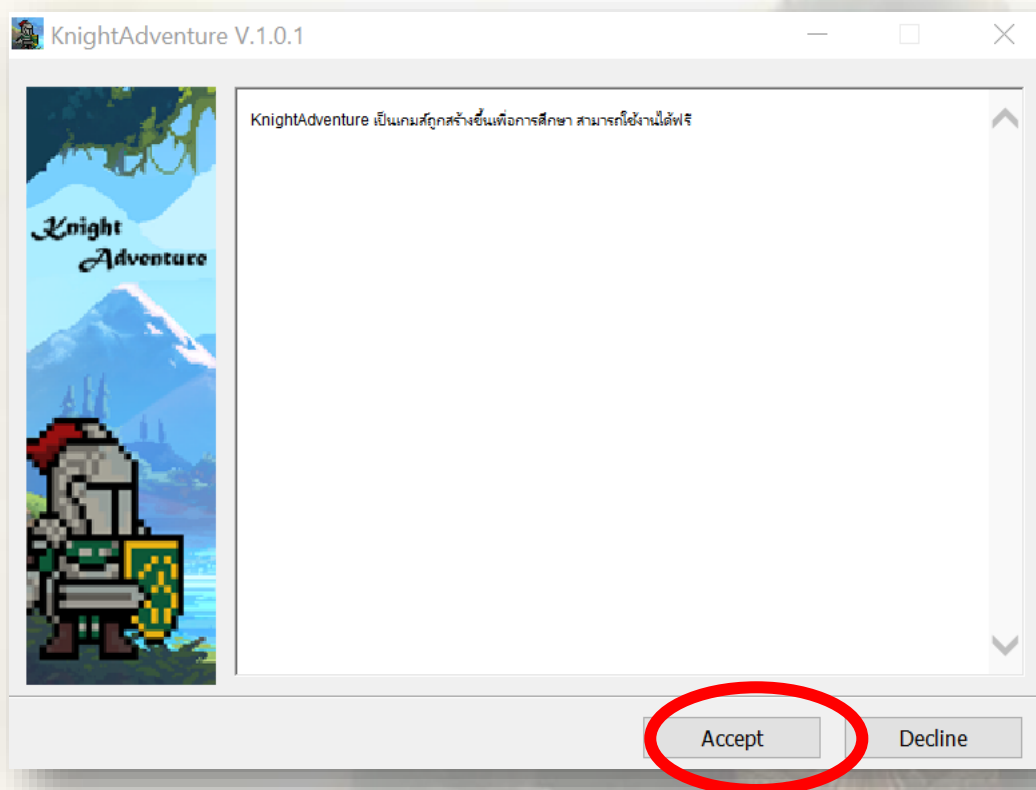


Program Installation

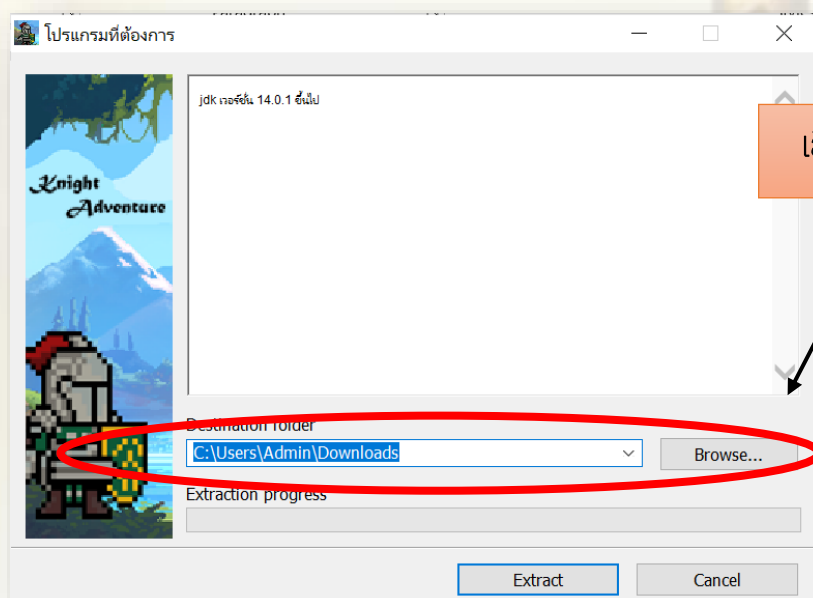
วิธีที่ 1

ขั้นตอนที่ 1 : Download ตัวไฟล์ setup เกมจากลิงก์ <https://bit.ly/31xVAH5>

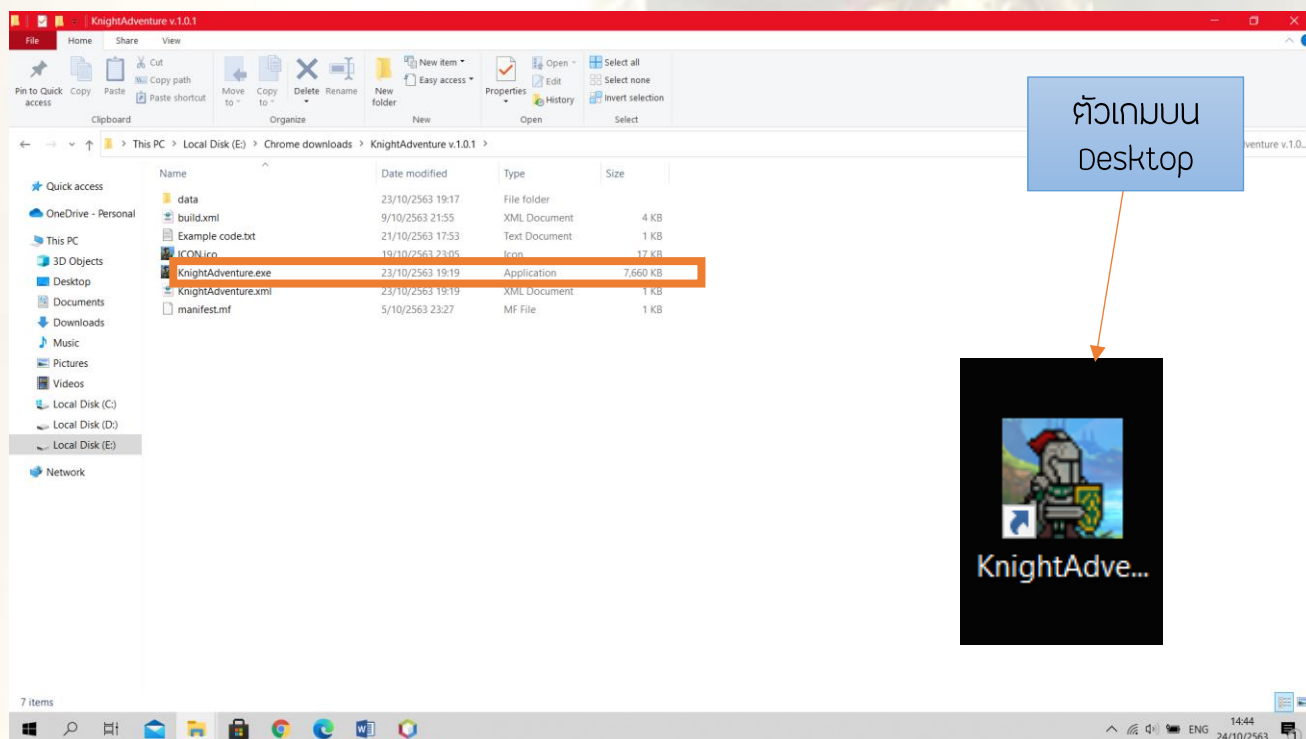
ขั้นตอนที่ 2 : จากนั้นเปิดตัวไฟล์ที่ได้มา จะได้หน้าต่างดังนี้ จากนั้นกด accept



ขั้นตอนที่ 3 : เลือกไฟล์คอร์ที่ทำการจะลงเกมเอาไว้ จากนั้นกด Extract



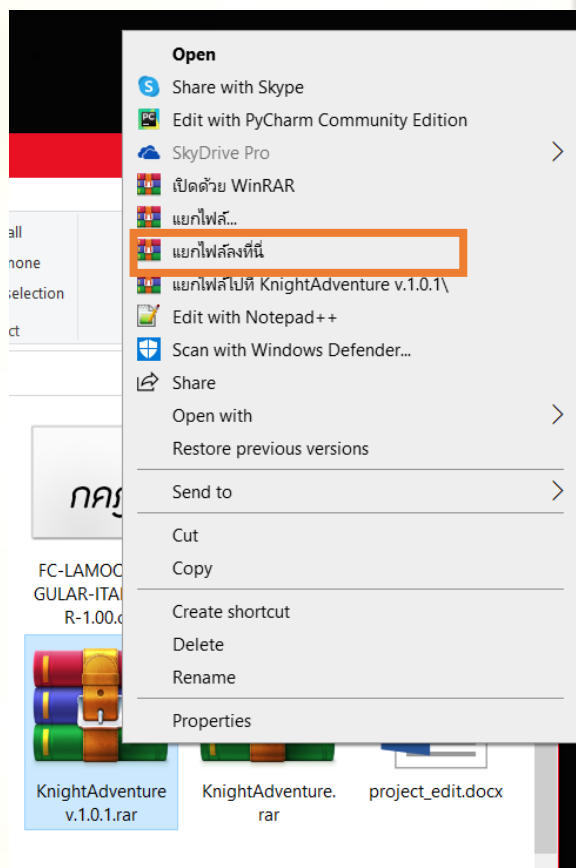
ขั้นตอนที่ 4 : ไปยังไฟล์คอร์ที่ได้ลงเกมเอาไว้ กดตัวเกมแล้วเข้าเล่นได้ทันทีหรือเราสามารถเข้าเล่นได้จาก icon ที่หน้า desktop ได้เลย



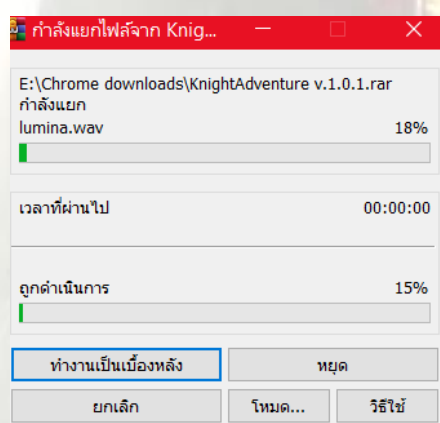
วิธีที่ 2

ขั้นตอนที่ 1 : ดาวน์โหลดไฟล์ rar จากลิงก์ <https://rb.gy/j75iw8> เพื่อนำไปแตกไฟล์

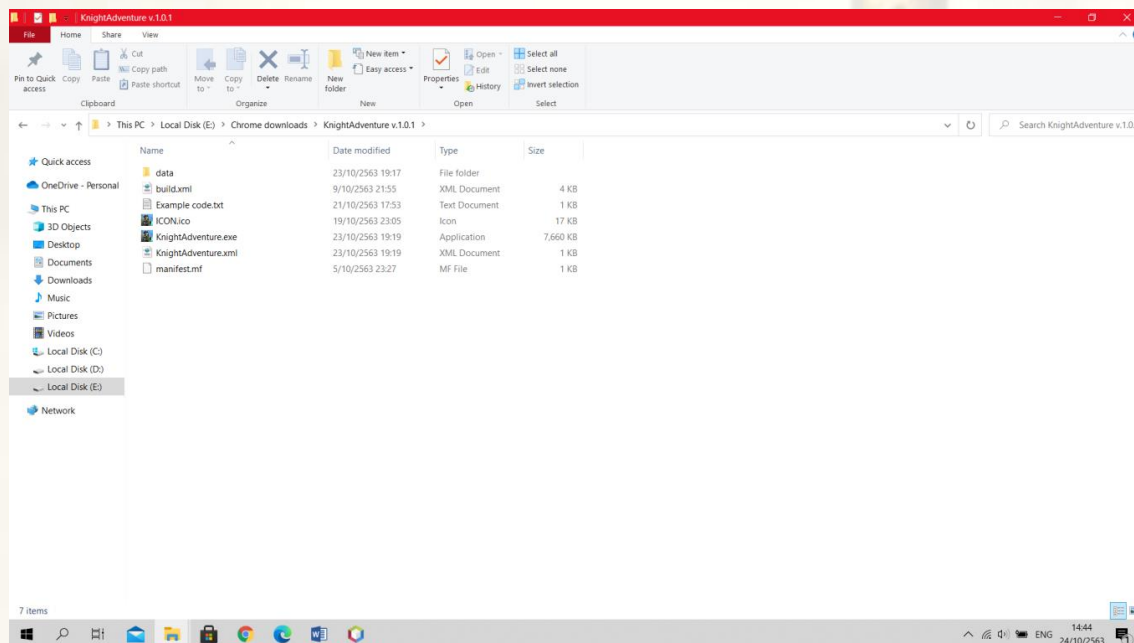
ขั้นตอนที่ 2 : ทำการแตกไฟล์โดยการคลิกขวาที่ไฟล์ จากนั้นคลิกแยกไฟล์ลงที่นี่ หรือ Extract Here



ขั้นตอนที่ 3 : รอการแตกไฟล์ เมื่อแตกไฟล์เสร็จสามารถเข้าเล่นเกมได้ทันที



ขั้นตอนที่ 4 : จะได้ไฟล์เคอร์เกมขึ้นมา คลิกเข้าไปในไฟล์เคอร์สามารถเข้าเกมได้ทันที แต่วิธีนี้จะไม่มีตัว shortcut บนหน้า Desktop ให้นั่นเอง



วิธีแก้ปัญหาเมื่อไม่สามารถเข้าเกมได้

หากผู้เล่นไม่สามารถเข้าเกมได้ ให้ผู้เล่น Download JDK เวอร์ชัน 14.0.1 จากลิงก์นี้ไปติดตั้งนะ ครับ rb.gy/iuj2sd

Reference

- ✚ <http://code.function.in.th/java-base> : การสอนภาษา java เบื้องต้น
- ✚ https://www.youtube.com/watch?v=y4dfsushUaw&list=LLAJAHyHstivOInipUDV_pJQ&index=1&ab_channel=prasertcbs : วิดีโอสอนการอ่านและเขียนเท็กซ์ไฟล์
- ✚ https://www.youtube.com/watch?v=5kTh-dmnBWo&t=1s&ab_channel=mcProgramming : สอนการเขียนโปรแกรม java multithreding กับ Runnable
- ✚ https://www.youtube.com/watch?v=iFovXtvunGE&ab_channel=BroCode : การเขียน KeyListener
- ✚ https://www.youtube.com/watch?v=xWONp8JvO4g&t=363s&ab_channel=KnowledgeToShare : สอนการเขียน java.swing GUI สำหรับการเขียนเกม
- ✚ https://www.youtube.com/watch?v=RudWnkkkWCbm&ab_channel=SwSosweet : การใส่ภาพพื้นหลังใน GUI
- ✚ https://www.youtube.com/watch?v=xs92kqU2yUwg&list=PLRXOVpVlEmdDySO3wDqmyGKMVH4Qa4QhR&ab_channel=%EB%8F%99%EB%B9%88%EB%82%98 : เขียนโปรแกรม java สำหรับสร้างเกมเบื้องต้น
- ✚ <https://www.geeksforgeeks.org/java-math-random-method-examples/> : การใช้ random ในภาษา java
- ✚ https://www.youtube.com/watch?v=F7un4UWXDTg&ab_channel=PRASITTECH : การแปลงไฟล์จากตัว .java เป็น .exe โดยโปรแกรม Lunch4j
- ✚ <https://www.thaicreate.com/java/java-create-dir-delete-directory.html> : วิธีการลบไฟล์จากโฟลเดอร์โดยภาษา java
- ✚ https://www.youtube.com/watch?v=kkrchlwidps&ab_channel=%E0%B9%84%E0%B8%9E%E0%B8%A3%E0%B8%A7%E0%B8%B1%E0%B8%A5%E0%B8%A2%E0%B9%8C%E0%B8%81%E0%B8%A5%E0%B9%88%E0%B8%AD%E0%B8%A1%E0%B8%88%E0%B8%B4%E0%B8%95%E0%B8%A3%E0%B9%8C : การสร้างตัว Setup เพื่อนติดตั้งเกม
- ✚ <https://online-audio-converter.com/th/> : ตัวแปลงไฟล์เสียง

+ <https://convertio.co/th/> : แปลงไฟล์ไปเป็นไฟล์ใดก็ได้

รูปภาพประกอบภายในเกม

+ <https://www.gameart2d.com/the-volcano---top-down-tileset.html>

+ https://www.pinterest.com/pin/112590059418211071/?nic_v2=1a7wx5al5

+ https://www.reddit.com/r/PixelArt/comments/7mxeo8/cc_how_can_i_improve_my_first_attempt_at_a_grass/

+ <https://www.dreamstime.com/illustration/lava-monster.html>

+ <https://www.pinterest.com/pin/593278950887815405/>

+ <https://www.pngegg.com/en/png-eeest>

+ <https://www.pinterest.com/pin/82472236911007301/>

+ <https://www.pinterest.com/pin/361836151311168309/>

+ <https://www.pinterest.com/pin/467459636324548022/>

+ <https://www.artstation.com/marketplace/p/d0kk/game-asset-2d-animated-fantasy-skeletons>

+ <https://www.shutterstock.com/th/image-illustration/pixel-art-bone-dinosaur-skeleton-8-1439346500>

+ <https://www.pixilart.com/art/pixel-dirt-1de6caa03153a9b>

+ <https://www.pixilart.com/art/pixel-grass-block-f2a5b7831a9ca58>