

Running Dino!

06016317 Object Oriented Programming

CONTENTS

1 รายละเอียดของเกม

กี่มีของเกม, วัตถุประสงค์, ประโยชน์

2 Interface ของเกม

หน้าต่างภายในเกม

3 ไดอาแกรม

ภาพรวมทั้งระบบของเกม



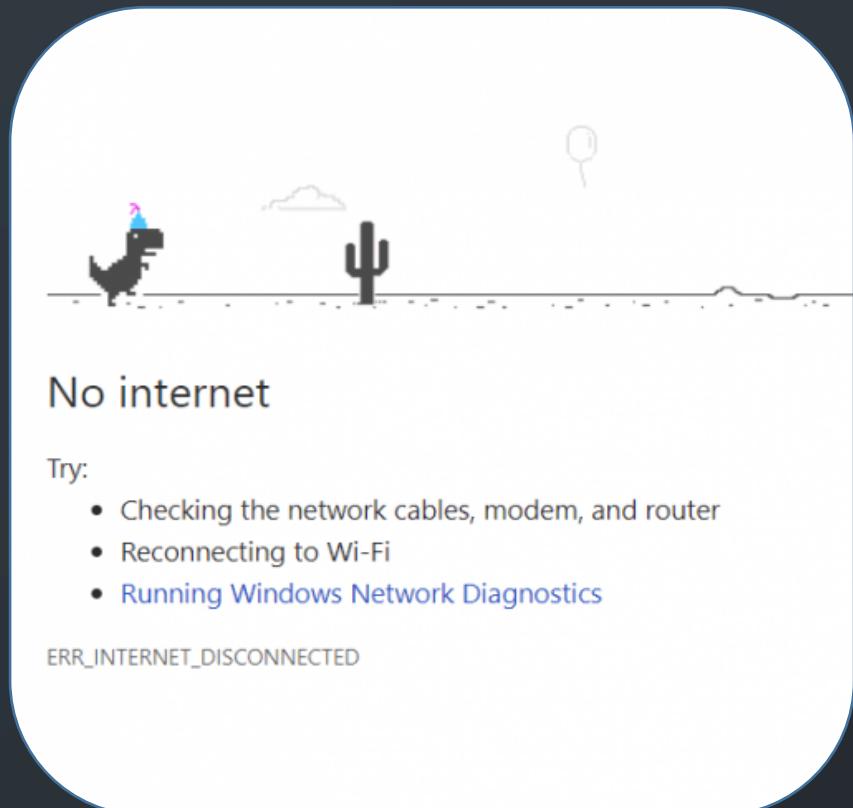
01

PART ONE

รายละเอียดของเกม

รายละเอียดของเกม

ที่มาของเกม



กลุ่มของเราได้รับแรงบันดาลใจมาจากเกมไดโนเสาร์กระโดดผ่านสิ่งกีดขวางที่อยู่บนเบราว์เซอร์ จึงมีแนวคิดที่อยากพัฒนาเกมไดโนเสาร์กระโดดผ่านสิ่งกีดขวางนี้ให้มีความน่าสนใจมากยิ่งขึ้น โดยมีชื่อเกมว่า “Running Dino!” ซึ่งพัฒนามาจากภาษา Java ในรูปแบบของ Object Oriented Programming โดยมีจุดเด่นคือมีดีไซน์ที่น่ารักสดใส มีฟังก์ชันที่ใช้งานได้ และสามารถเล่นได้ทุกแพลตฟอร์มทุกวัย



รายละเอียดของเกม

วัตถุประสงค์

- ▶ เพื่อสร้างความสนุกสนานและความเพลิดเพลินให้แก่ผู้เล่น
- ▶ เพื่อใช้เวลาให้เกิดประโยชน์สูงสุด
- ▶ เพื่อฝึกหลักการเขียน Object Oriented Programming
- ▶ เพื่อฝึกพื้นฐานการเขียนภาษา Java

รายละเอียดของเกม

ประโยชน์ที่ได้รับ

ได้ฝึกสมาร์ตมากยิ่งขึ้นทำให้มีความจดจำกับสิ่งที่

ทำอยู่ได้มากขึ้น

ได้ฝึกการทำงานร่วมกับผู้อื่น

ได้รับความสนุกสนานและความเพลิดเพลิน

ได้เรียนรู้วิธีการสร้างและพัฒนาเกมโดยใช้ Object
Oriented Programming ด้วยภาษา Java

ได้ใช้เวลาว่างให้เกิดประโยชน์สูงสุด

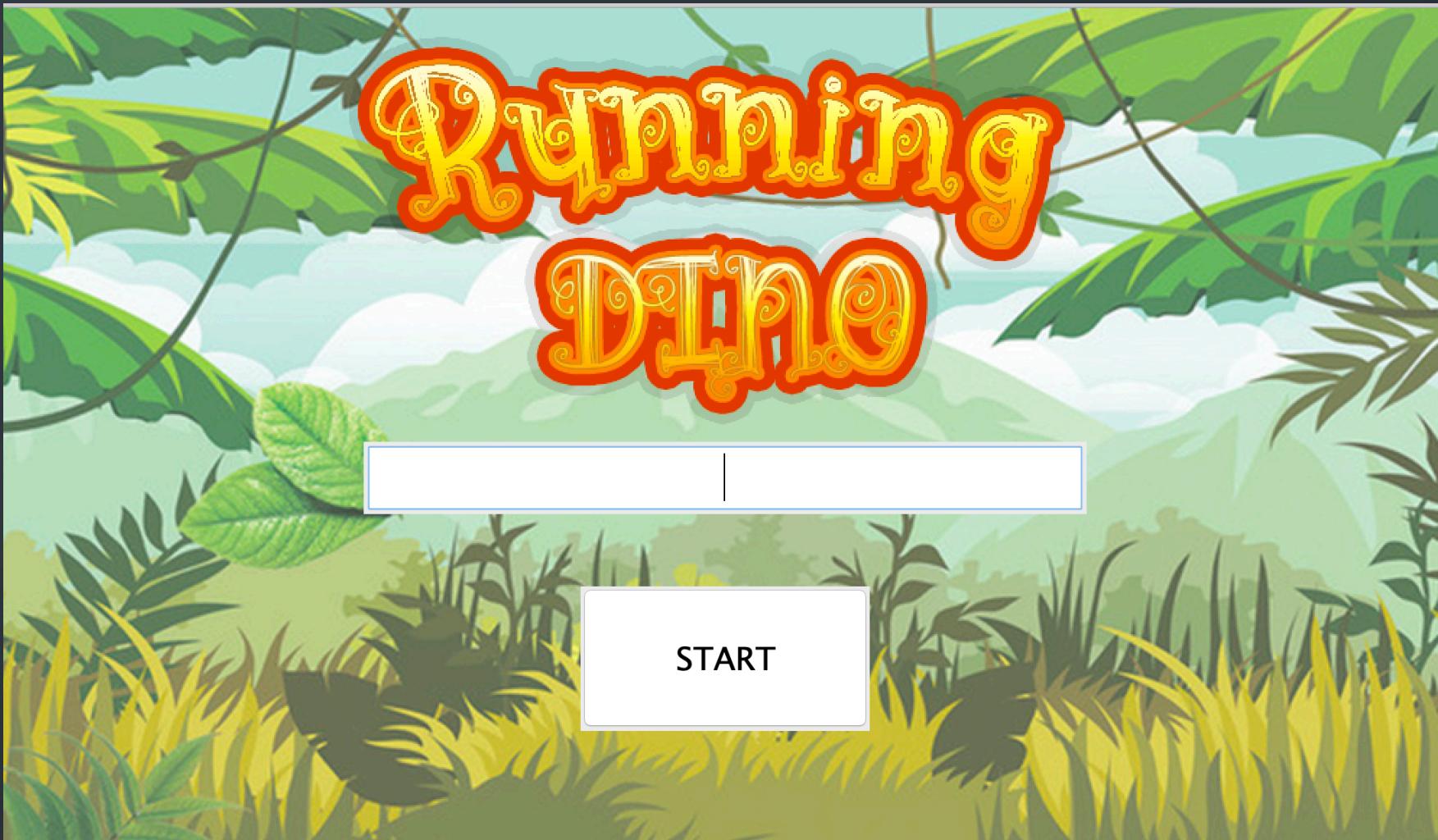


02

PART TWO

Interface ของเกม

หน้าแรกของเกม



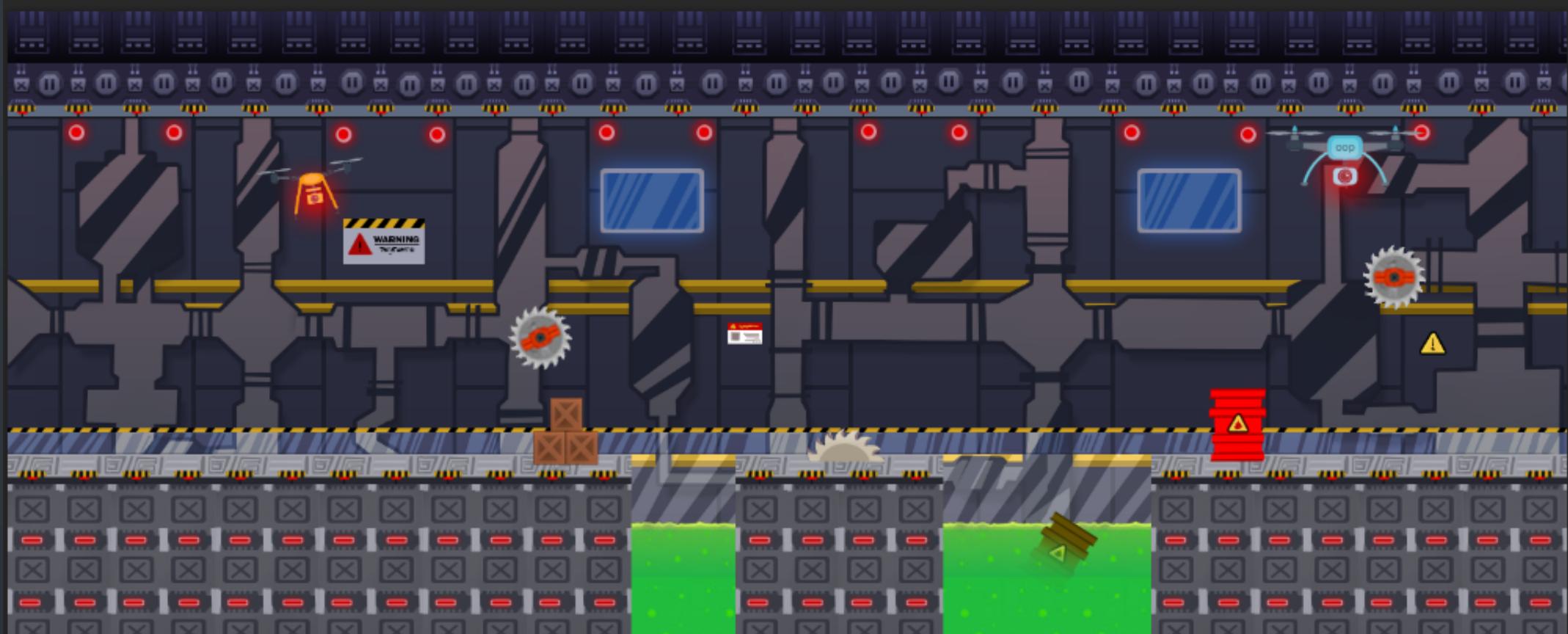
• ด่านต่าง ๆ ภายในเกม •



• ด่านต่าง ๆ ภายในเกม •



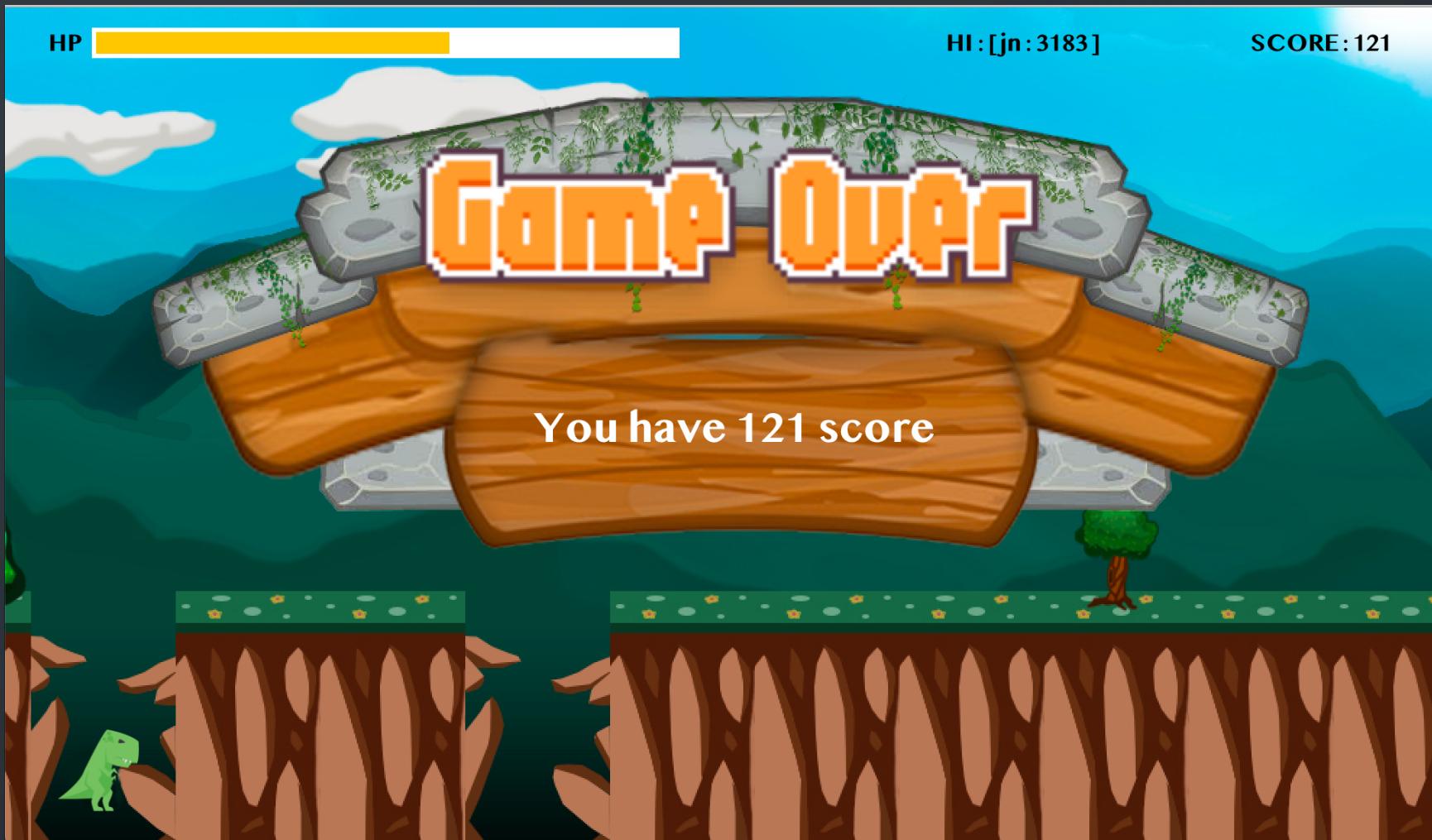
• ด่านต่าง ๆ ภายในเกม •



• ด่านต่อไป ภายในเกม •



หน้าเกมโวเวอร์





03

PART THREE

ไดอาแกรม

DinoCharacter	
+ score	: int
+ LAND_POSY	: int
+ GRAVITY	: float
- NORMAL_RUN	: int
- JUMPING	: int
- DOWN_RUN	: int
- DEATH	: int
- posY	: float
- posX	: float
- speedX	: double
- speedY	: float
- rectBound	: Rectangle
- countJump	: int
- hp	: int
- state	: NORMAL_RUN
- normalRunAnim	: Animation
- jumping	: BufferedImage
- downRunAnim	: Animation
- deathImage	: BufferedImage
- jumpSound	: AudioClip
- deadSound	: AudioClip
- scoreJpSound	: AudioClip
- normalState	: BufferedImage
- downState	: BufferedImage
+ DinoCharacter ()	
+ getSpeedX ()	: double
+ setSpeedX (double speedX)	: void
+ draw (Graphics g)	: void
+ update ()	: void
+ jump ()	: void
+ down (boolean isDown)	: void
+ getBound ()	: Rectangle
+ dead (boolean isDeath)	: void
+ reset ()	: void
+ playDeadSound ()	: void
+ playScoreSound ()	: void
+ setLAND_POSY (int g)	: void
+ getLAND_POSY ()	: int
+ getPosX	: float
+ getDinoWidth	: float
+ getPosY ()	: float
+ getIp	: int
+ setIp	: void

RunnerScore	
+ upSpeedGame	: int
+ run ()	: void

Resource	
+ getResouceImage (String path)	: BufferedImage

ObjectGameManager	
+ countEnemy	: int
+ LAND_POSY	: int
- listLand	: List<landBox>
- land[]	: BufferedImage
- pitCount	: int
- mainCharacter	: DinoCharacter
- pit.list[]	: BufferedImage
- rand	: Random
- listEnemies	: ArrayList<Enemy>
- boxWidth	: int
- blankBox	: blankBox
- pre	: landBox
+ ObjectGameManager (int width, DinoCharacter mainCharacter)	
+ createEnemy (int type, int posX)	: Enemy
+ draw (Graphics g)	: void
+ update ()	: void
+ randomNumber (int limit)	: int
+ isCollision ()	: boolean
+ newStage ()	: void
+ reset ()	: void
- setImageLand (landBox imgLand, int type)	: void

HighScoreStorage	
- name	: String
- hightscore	: int
+ getHightscore ()	: int
+ getName ()	: String
+ saveHightscore (int hs)	: void
+ saveName (String n)	: void

GameWindow	
+ SCREEN_WIDTH	: int
- gameScreen	: GameScreen
+ GameWindow ()	
+ startGame (String n)	: void

landBox	
+ posX	: float
+ image	: BufferedImage
+ numberOftPit	: int
+ keyReleased (KeyEvent e)	: void
+ keyTyped (KeyEvent e)	: void
- resetGame ()	: void
+ setSpeed (int d)	: void

Clouds	
- listCloud	: List<ImageCloud>
- cloud[]	: BufferedImage
- mainCharacter	: DinoCharacter
+ Clouds (int width, DinoCharacter mainCharacter)	
+ update ()	: void
+ draw (Graphics g)	: void

GameScreen	
+ enemyAndLandCount	: int
- blankBox	: boolean
- countStage**	: int
- name	: String
- namshs	: String
- hightscore	: int
- jumpcount	: int
- bg	: BufferedImage
- gameOver	: BufferedImage
- backgroundPoint	: double
- START_GAME_STATE	: int
- GAME_PLAYING_STATE	: int
- GAME_OVER_STATE	: int
- manager	: ObjectGameManager
- mainCharacter	: DinoCharacter
- clouds	: Clouds
- thread	: Thread
- score	: RunnerScore
- runScore	: Thread
- isKeyPressed	: Boolean
+ enemyAndLandCount	: int
- gameState	: int
- replayButtonImage	: BufferedImage
- gameOverButtonImage	: BufferedImage
- speedGameM	: int
- speedGameN	: int
+ GameScreen ()	
+ getNameFirst (String n)	: void
+ startGame ()	: void
+ gameUpdate ()	: void
+ paint (Graphics g)	: void
+ run ()	: void
+ getCountStage ()	: int
+ keyReleased (KeyEvent e)	: void
+ keyTyped (KeyEvent e)	: void
- resetGame ()	: void
+ update (double posX)	: void
+ draw (Graphics g)	: void
+ getBound	: Rectangle
+ getPosX	: double
- randomLandY ()	: int

StartWindow	
+ SCREEN_WIDTH	: int
- background	: BufferedImage
- head1	: BufferedImage
- head2	: BufferedImage
- name	: String
- gameWindow	: GameWindow
- sp	: JPanel
- n	: JTextField
- start	: JButton
- screenSize	: Dimension
+ StartWindow ()	
+ paint (Graphics g)	: void
+ startPage ()	: void
+ main (String args [])	: void
+ actionPerformed (ActionEvent e)	: void

Animation	
- list	: List<BufferedImage>
- deltaTime	: long
- currentFrame	: int
- previousTime	: long
+ Animation (int deltaTime)	
+ updateFrame ()	: void
+ addFrame (BufferedImage image)	: void
+ getFrame ()	: BufferedImage

AirEnemy	
+ Y_LAND	: int
- posX	: double
- width	: int
- height	: int
- image	: BufferedImage
- mainCharacter	: DinoCharacter
- enemyCounts	: int
- rectBound	: Rectangle
- listCactus []	: bufferedImage
- flyAnim	: Animation
+ AirEnemy (DinoCharacter mainCharacter, int posX)	
+ update (double posX)	: void
+ draw (Graphics g)	: void
+ getBound	: Rectangle
+ getPosX	: double
- randomLandY ()	: int

blankBox	
- posX	: int
+ blankBox (DinoCharacter mainCharacter, int posX)	
+ update (double posx)	: void

Enemy*	
+ update (double posx)	: void
+ draw (Graphics g)	: void
+ getBound ()	: Rectangle
+ isOutOfScreen ()	: boolean

สมาชิกกลุ่ม

61070017 นายกิตติวัชร เอี่ยมกิจการ

61070030 นางสาวชฎานี คำเจริญ

61070037 นางสาวธิรนรัตน์ บูรณะพิสิฐ

61070156 นายภักรนันท์ เรืองชนะ

61070163 นางสาวภาณี ทองบ่อ