Langkah Langkah Pembuatan Project Game Greenfoot

Monster Madness

Oleh :

Panji Arief Pratama

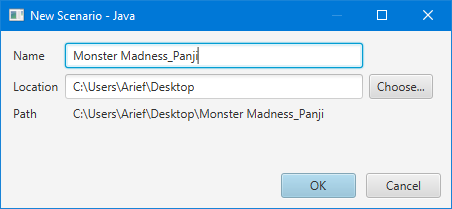
21041001

POLITEKNIK HARAPAN BERSAMA TEGAL

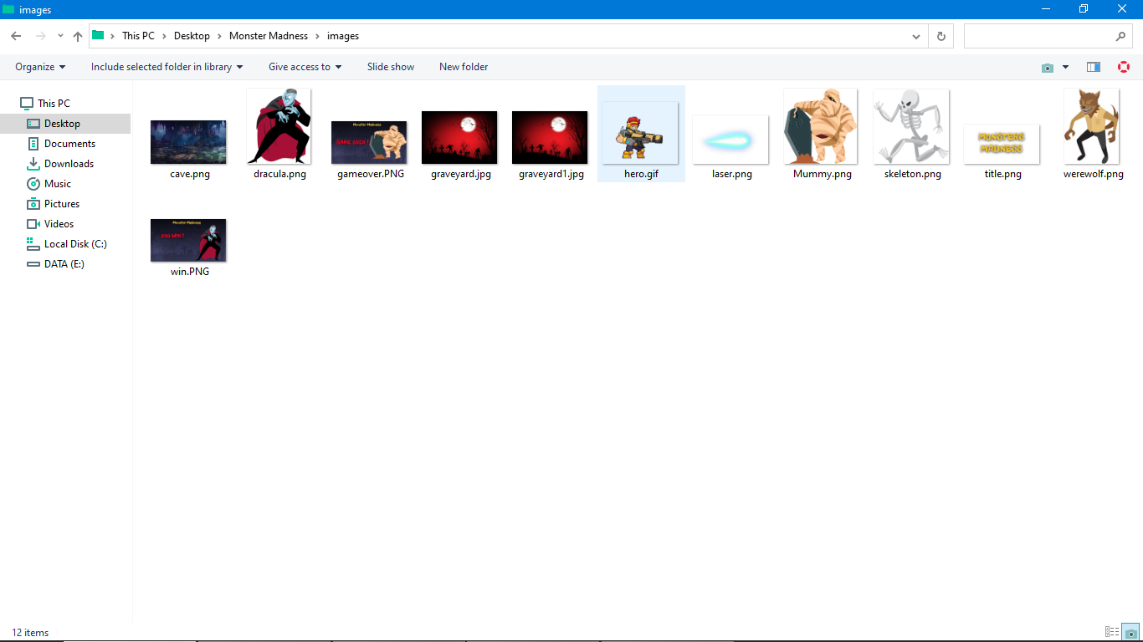
2023

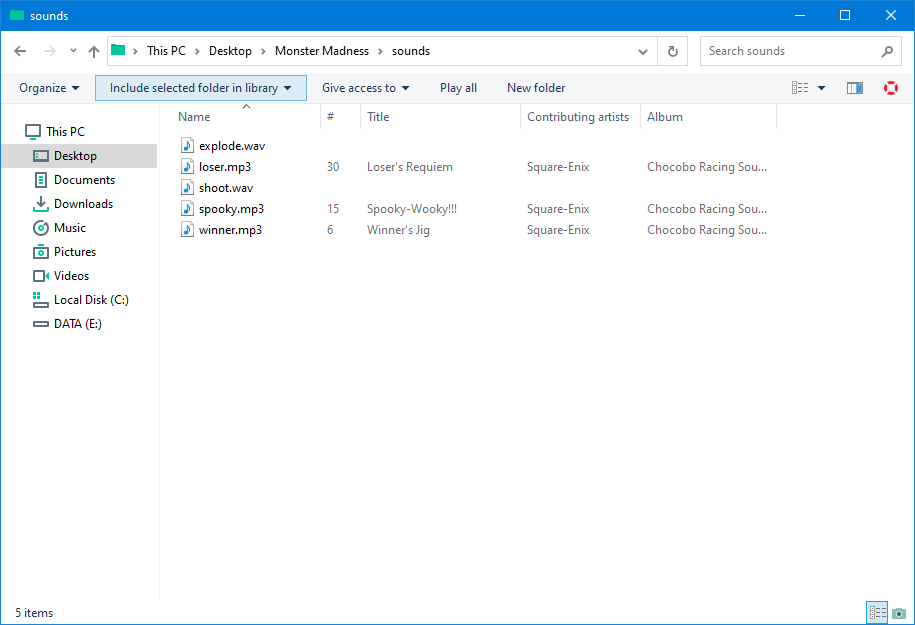
# **Membuat Project Greenfoot**

1. Membuat project baru pada greenfoot (java scenario) dengan nama ‘Monster Madness’.

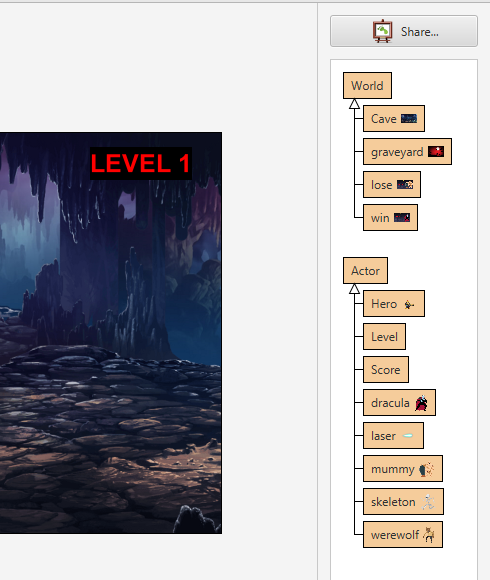


1. Menyiapkan semua gambar dan suara yang akan digunakan dalam project.





1. Memasukkan semua world dan actor yang akan digunakan, menyesuaikan dengan gambar yang telah disiapkan sebelumnya.



# **Class Hero**

1. Double klik pada Actor ‘Hero’, dan ketikan code berikut kedalam Class Hero.

import greenfoot.\*; // (World, Actor, GreenfootImage, Greenfoot and MouseInfo)

public class Hero extends Actor

{

private int speed = 4; //satuan gerakan hero

private final int GRAVITY = 1; // satuan gerakan vertikal hero

private int velocity; // kecepatan hero

boolean canFire; //cek penembakan

public Hero()

{

velocity = 0;

}

public void act()

{

fall();

move();

fireProjectile();

attacked();

}

// method untuk menggerakan hero

public void move()

{

if (Greenfoot.isKeyDown("left"))

{

setLocation (getX() - speed, getY());

}

if (Greenfoot.isKeyDown("right"))

{

setLocation (getX() + speed, getY());

}

if (Greenfoot.isKeyDown("up")&& getY() > getWorld().getHeight()-100)

{

jump();

}

}

//method hero jatuh

public void fall()

{

setLocation (getX(), getY() + velocity);

if (getY() > getWorld().getHeight() -100) velocity = 0;

else velocity += GRAVITY;

}

//method hero lompat

public void jump()

{

velocity = -12;

fall();

}

//method tembak laser

public void fireProjectile()

{

if(Greenfoot.isKeyDown("space") && canFire == true)

{

getWorld().addObject(new laser(), getX() +100 , getY()+20);

canFire = false;

Greenfoot.playSound("shoot.wav");

}

else if (!Greenfoot.isKeyDown("space"))

{

canFire = true;

}

}

//method saat hero terkena monster

public void attacked()

{

werewolf B = (werewolf)getOneIntersectingObject(werewolf.class);

if(B!=null)

{

Greenfoot.setWorld(new lose());

}

dracula C = (dracula)getOneIntersectingObject(dracula.class);

if(C!=null)

{

Greenfoot.setWorld(new lose());

}

skeleton D = (skeleton)getOneIntersectingObject(skeleton.class);

if(D!=null)

{

Greenfoot.setWorld(new lose());

}

mummy E = (mummy)getOneIntersectingObject(mummy.class);

if(E!=null)

{

Greenfoot.setWorld(new lose());

}

}

}

# **Monster**

1. Monster pada Level 1 diberi nama dengan **werewolf**, double klik pada Actor **werewolf** dan masukan code berikut ke dalam Class **werewolf**.

import greenfoot.\*; // (World, Actor, GreenfootImage, Greenfoot and MouseInfo)

public class werewolf extends Actor

{

public void act()

{

move();

getHit();

}

//method gerakan

public void move()

{

setLocation(getX()-5, getY());

}

//method ketika terkena laser

public void getHit()

{

laser A = (laser)getOneIntersectingObject(laser.class);

if(A!=null)

{

getWorld().removeObject(A);

World world = getWorld();

Cave myWorld = (Cave)world;

Score score = myWorld.getScore();

score.addScore();

getWorld().removeObject(this);

}

}

}

1. Monster pada Level 2 diberi nama dengan **skeleton**, double klik pada Actor **skeleton** dan masukan code berikut ke dalam Class **skeleton**.

import greenfoot.\*; // (World, Actor, GreenfootImage, Greenfoot and MouseInfo)

public class skeleton extends Actor

{

public skeleton()

{

GreenfootImage image = getImage();

image.scale(image.getWidth()-50, image.getHeight()-50);

setImage(image);

}

public void act()

{

move();

getHit();

}

//method gerakan

public void move()

{

setLocation(getX()-4, getY());

}

//function ketika terkena laser

public void getHit()

{

laser A = (laser)getOneIntersectingObject(laser.class);

if(A!=null)

{

getWorld().removeObject(A);

World world = getWorld();

graveyard myWorld = (graveyard)world;

Score score = myWorld.getScore();

score.addScore();

getWorld().removeObject(this);

}

}

}

# **Boss Monster**

1. Boss Monster pada Level 1 diberi nama dengan **dracula**, double klik pada Actor **dracula** dan masukan code berikut ke dalam Class **dracula**.

import greenfoot.\*; // (World, Actor, GreenfootImage, Greenfoot and MouseInfo)

public class dracula extends Actor

{

int timesHit = 10; //jumlah tembakan yang diperlukan

public dracula()

{

GreenfootImage image = getImage();

image.scale(image.getWidth() / 2, image.getHeight() -100);

setImage(image);

}

public void act()

{

move();

getHit();

}

//method gerakan

public void move()

{

setLocation(getX()-2, getY());

}

//method ketika terkena laser

public void getHit()

{

laser A = (laser)getOneIntersectingObject(laser.class);

if(A!=null)

{

getWorld().removeObject(A);

World world = getWorld();

Cave myWorld = (Cave)world;

Score score = myWorld.getScore();

score.addScore();

timesHit--;

}

if (timesHit == 0)

{

getWorld().removeObject(this);

}

}

}

1. Boss Monster pada Level 2 diberi nama dengan **mummy**, double klik pada Actor **mummy** dan masukan code berikut ke dalam Class **mummy**.

import greenfoot.\*; // (World, Actor, GreenfootImage, Greenfoot and MouseInfo)

public class mummy extends Actor

{

int timesHit = 20; //jumlah tembakan yang diperlukan

public mummy()

{

GreenfootImage image = getImage();

image.scale(image.getWidth(), image.getHeight() -50);

setImage(image);

}

public void act()

{

move();

getHit();

}

//method gerakan

public void move()

{

setLocation(getX()-1, getY());

}

//method ketika terkena laser

public void getHit()

{

laser A = (laser)getOneIntersectingObject(laser.class);

if(A!=null)

{

getWorld().removeObject(A);

World world = getWorld();

graveyard myWorld = (graveyard)world;

Score score = myWorld.getScore();

score.addScore();

timesHit--;

}

if (timesHit == 0)

{

getWorld().removeObject(this);

}

}

}

# **Class World**

1. World pada Level 1 diberi nama dengan world **Cave,** world ini yang akan dimunculkan saat pertama kali game dijalankan. Double klik pada world **Cave,** dan masukkan code berikut pada world **Cave.**

import greenfoot.\*; // (World, Actor, GreenfootImage, Greenfoot and MouseInfo)

public class Cave extends World

{

private Score score; //inisiasi score di world cave

private Hero hero; //inisiasi hero

private Level level; //inisiasi status level saat ini

boolean bossLevel = false; //inisiasi akan munculnya boss monster pada level ini

public Cave()

{

super(1000, 400, 1);

prepare();

}

public void act ()

{

Boss1();

Enemy1();

nextLevel();

}

//method menampilkan nilai score

public Score getScore()

{

return score;

}

//method menampilkan status level

public Level getLevel()

{

return level;

}

//method memanggil hero dan score

public void prepare()

{

hero = new Hero();

addObject(hero,100, 300);

score = new Score();

addObject(score, 80, 30);

level = new Level();

addObject(level, 920, 30);

}

//method memanggil monster di level 1

public void Enemy1()

{

if(Greenfoot.getRandomNumber (50)<1)

{

addObject(new werewolf(), 950 , 320);

}

}

//method memunculkan boss level 1

public void Boss1()

{

if(score.counter == 10 || score.counter == 11)

{

if(bossLevel == false)

{

addObject(new dracula(), 950 , 320);

bossLevel = true;

}

}

}

//method untuk beralih ke level selanjutnya

public void nextLevel()

{

if(score.counter >=11 &&getObjects(dracula.class).size()== 0)

{

level.addLevel();

Greenfoot.setWorld(new graveyard( hero, score, level)); /\*\*parameter hero, score, dan level

agar dapat dipanggil di world yang lain

dengan nilai yang sama\*\*/

}

}

}

1. World pada Level 2 diberi nama dengan world **graveyard,** world ini akan muncul saat Hero telah berhasil menghancurkan Boss Monster Level 1. Double klik pada world **graveryard,** dan masukkan code berikut pada world **graveyard.**

import greenfoot.\*; // (World, Actor, GreenfootImage, Greenfoot and MouseInfo)

public class graveyard extends World

{

private Hero hero; //inisiasi hero

private Score score; //inisiasi score

private Level level;

boolean bossLevel = false; //inisiasi bossLevel

public graveyard(Hero h, Score s, Level l) //memanggil hero dan score di level sebelumnya

{

super(1000, 400, 1);

this.hero = h; //memanggil hero yang sama dengan world sebelumnya

this.score = s; //memanggil score yang sama dengan world sebelumnya

this.level = l; //memanggil status level

prepare();

}

//method untuk menambah nilai score

public Score getScore()

{

return score;

}

public void act()

{

Enemy2();

Boss2();

nextLevel();

}

public void prepare()

{

addObject(hero,100, 300);

addObject(score, 80, 30);

addObject(level, 920, 30);

}

//method memanggil monster di level 2

public void Enemy2()

{

if(Greenfoot.getRandomNumber (30)<1)

{

addObject(new skeleton(), 950 , 320);

}

}

//method memunculkan boss level 2

public void Boss2()

{

if(score.counter == 40 || score.counter == 41)

{

if(bossLevel == false)

{

addObject(new mummy(), 950 , 320);

bossLevel = true;

}

}

}

public void nextLevel()

{

if(score.counter >60 && getObjects(mummy.class).size() == 0)

{

Greenfoot.setWorld(new win(score)); /\*\*parameter score,agar dapat dipanggil pada world sebelumnya dengan nilai yang sama\*\*/

}

}

}

1. World **win,** ditampilkan saat Hero berhasil menghancurkan Boss Monster Level 2 dan akan memunculkan hasil score yang diperoleh. Double klik pada world **win,** dan masukkan code berikut pada world **win.**

import greenfoot.\*; // (World, Actor, GreenfootImage, Greenfoot and MouseInfo)

public class win extends World

{

private Score score; //inisiasi variable score

public win(Score s) /\*parameter score yang dibawa dari world sebelumnya,

agar nilainya sama dan dapat ditampilkan.\*/

{

super(920, 520, 1);

this.score = s; //memanggil score pada world sebelumnya

addObject(score, 300, 300);

Greenfoot.playSound("winner.mp3");

}

}

1. World **lose**, ditampilkan saat Moster atupun Boss Monster mengenai Hero, dan permainan akan berakhir. Double klik pada world **lose,** dan masukkan code berikut pada world **lose.**

import greenfoot.\*; // (World, Actor, GreenfootImage, Greenfoot and MouseInfo)

public class lose extends World

{

public lose()

{

super(920, 520, 1);

Greenfoot.playSound("loser.mp3");

}

}

# **Class Laser, Score dan Level**

1. Class **laser, laser** ditembakkan oleh Hero saat player menekan tombol ‘space’ sebagaimana telah dinyatakan dalam **fireProjectile()** atau method tembak laser dalam Class Hero. Dalam class **laser** ini berisi kecepatan laser saat ditembakan dan hapus laser saat mencapai batas tertentu. Double klik pada Class **laser,** dan masukkan code berikut.

import greenfoot.\*; // (World, Actor, GreenfootImage, Greenfoot and MouseInfo)

public class laser extends Actor

{

public void act()

{

setLocation (getX()+5, getY());

if (getX()== 950)

{

getWorld().removeObject(this);

}

}

}

1. Class **Score,** setiap tembakan laser dari Hero yang mengenai Monster maupun Boss Monster akan menambahkan nilai score sebanyak +1. Double klik pada Class **Score,** dan masukkan code berikut.

import greenfoot.\*; // (World, Actor, GreenfootImage, Greenfoot and MouseInfo)

public class Score extends Actor

{

int counter = 0; //inisiasi nilai awal score

public Score()

{

setImage(new GreenfootImage("SCORE : " +counter,25,Color.YELLOW,Color.BLACK));

}

public void act()

{

setImage(new GreenfootImage("SCORE : " +counter,25,Color.YELLOW,Color.BLACK));

}

//method menambah nilai score

public void addScore()

{

counter++;

}

}

1. Class **Level,** berisi mengenai status level yang sedang dimainkan oleh player. Saat Hero berhasil menghancurkan Boss Monster, maka akan menambah status level +1 (naik ke level selanjutnya). Double klik pada Class **Level,** dan masukkan code berikut.

import greenfoot.\*; // (World, Actor, GreenfootImage, Greenfoot and MouseInfo)

public class Level extends Actor

{

int counter = 1; //inisiasi nilai awal level

public Level()

{

setImage(new GreenfootImage("LEVEL " +counter,32,Color.RED,Color.BLACK));

}

public void act()

{

setImage(new GreenfootImage("LEVEL " +counter,32,Color.RED,Color.BLACK));

}

//method menambah nilai level

public void addLevel()

{

counter++;

}

}