**Project: Type Wizard**

This game is a Role-Playing-Game. Player is a wizard that can use spell by typing spell in different word. This game is Endless Mode that player can play until the wizard die, wizard has level and status that affected to his strength ,for the enemies it’s strength increasing by the level of the wizard too.



Figure 1 showing main menu. You can press any key to start the game.



Figure 2 showing map and user interface in game.



Figure 3 showing how wizard cast spell.



Figure 4 showing monster Yeti get hit by meteor skill.



Figure 5 showing monster Yeti effected by Freezing de-buff.



Figure 6 showing monster Yeti has fallen.

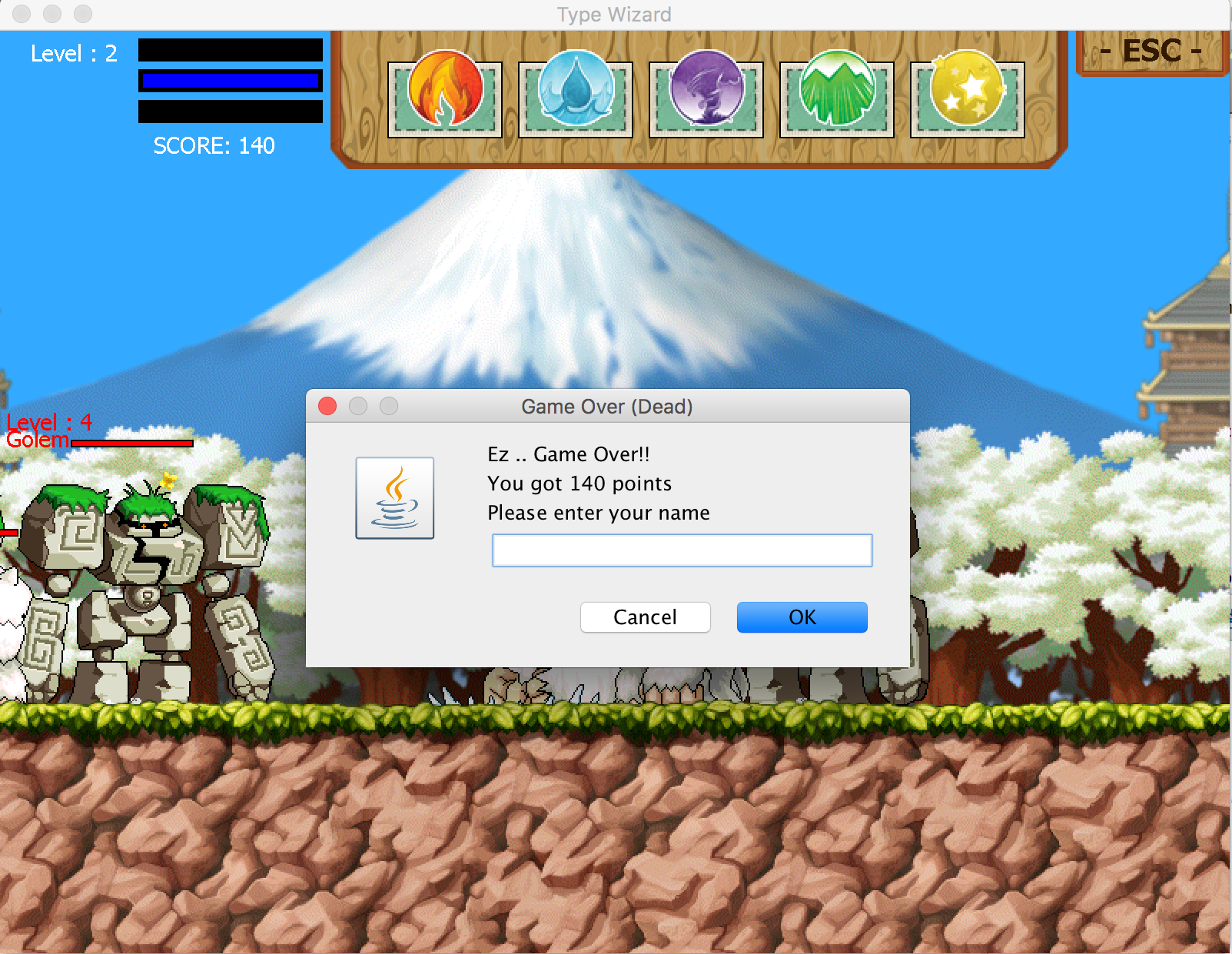


Figure 7 showing GameOver Screen.

**Package input**

**Class InputUtility**

Handle all of events occur during the game is running.

Field

* boolean[] keyPressed = new boolean[256];
* String spell = ""
* boolean escTriggered = false;
* boolean escPressed = false;

Constructor

* Default Constructor

Method

* boolean getKeyPressed(int key)
* void setKeyPressed (int key, boolean keyPressed)
* void clearSpell()
* void reset ()

**Package render**

**Class GameScreen**

…

Field

* long serialVersionUID

Constructor

GameScreen(); Initialize size and and KeyListener

Method

void paintComponent(Graphics g)

Class GameTitle

Field

boolean isTitle = true;

boolean swap = false;

int tick = 0;

int R = 255, G = 255, B = 255;

JLabel start = new JLabel("", JLabel.CENTER);

Constructor

GameTitle();

Method

void paintComponent(Graphics g)

void update()

Getter & Setter of isTitle and isSwap

Interface Renderable

void draw(Graphics g)

boolean isVisible()

Boolean getZ()

Class RenderableHolder

Field

RenderableHodler instance

List<Renderable> entities

Method

static RenderableHolder getInstance()

void add(Renderable o)

void remove(Renderable o)

void removeAll()

List<Renderable> getRenderableList()

Class Resource

Field

int screenWidth, screenHeight ;

Font wordFont,standardFont,biggerFont,pauseFont,skillFont;

BufferedImage hero, hero\_f, hero\_a, hero\_af, hero\_d, hero\_df ;

BufferedImage monster\_golem\_1\_1, monster\_golem\_1\_2, monster\_golem\_1\_3, monster\_golem\_1\_1\_f, monster\_golem\_1\_2\_f, monster\_golem\_1\_3\_f;

BufferedImage monster\_yeti\_1\_1, monster\_yeti\_1\_2, monster\_yeti\_1\_3, monster\_yeti\_1\_4, monster\_yeti\_1\_1\_f, monster\_yeti\_1\_2\_f, monster\_yeti\_1\_3\_f, monster\_yeti\_1\_4\_f;

BufferedImage statusBG, skillBoard, pauseBox, banner;

BufferedImage land1, background1, titleBG;

BufferedImage ice1, fire1, meteor1, meteor1\_2, poison1, spike1;

BufferedImage skillBoxBG, fireStatus, iceStatus, meteorStatus, poisonStatus, spikeStatus;

BufferedImage burn, poison, freeze;

AudioClip titlebgm, screenbgm, hit, jump, levelup, score, iceskill, meteorskill, meteorskill2, fireskill, poisonskill, spikeskill

Method

BufferedImage getImage(String directory)

AudioClip getSound(String directory)

Package entity

Abstract Class Moving

Field

-int x,y

Constructor

-Moving(int x, int y)

Method

-abstract void update()

Class Land

Field

-int xc

-boolean isStart,isEnd

Constructor

-Land(int x, int y)

Method

-void update()

-Getter & Setter

-void draw(Graphics2D g)

-boolean isVisible()

-int getZ()

Class Background

Field

int xc

-boolean isStart,isEnd

Constructor

-Background(int x, int y)

Method

-void update()

-Getter & Setter

-void draw(Graphics2D g)

-boolean isVisible()

-int getZ()

Class Hero

Field

int frameCount, count, countA, frameCountA, direction , temp, i , countD , frameCountD, tick, gravity, velocity, STR, INT, level, hp, attack, mana, manaTick , maxMp, maxHp, notBeHitCount’

boolean isJumped, isDead, isOutOfMana, isHitting, isRight, isLeft, isStop, isSkill, isCasting;

Land land;

Background background;

HeroStatus heroStatus;

Skill[] skills ;

Word[] words;

Monster nearMon

Constructor

Hero(int x,int y, Land land, Background background, HeroStatus heroStatus)

Method

void update()

void checkLevel();

void setHpMp();

void deadAnimation();

void idleSpellAnimation();

void idleAnimation();

void jump();

void walkLeft();

void walkRight();

void castingSkill();

void checkCastingSkill();

void removeMonster();

Getter & Setter

Class GameLogic

Field

Land land;

Hero hero;

HeroStatus heroStatus;

SkillStatus skillStatus;

boolean castFail , iCast, fCast, mCast, pCast, sCast, isPause ;

Word ice, fire, meteor, poison, spike;

int tick = 0, speed;

Background background;

Banner pauseBanner;

GameTitle title;

Thread t1,t2;

BgmControl bgm;

Constructor

GameLogic(GameTitle gameTitle)

Method

void update()

Getter & Setter

Class Word

Field

String[] words;

int[] x;

boolean[] press;

boolean isVisible;

Constructor

Word(String s)

Method

void draw(Graphics2D g);

boolean isEqual();

boolean cast(String s)

Getter & Setter

Class BgmControl

Field

boolean isPause;

Method

void run();

Getter & Setter

Package entity.skill

Abstract Class Skill

Field

boolean isPlaying;

int x, y, frameCount, count, frameWidth, frameHeight, attackRange,direction;

Method

abstract void update();

void play();

void stop();

int getZ();

setX();

Getter & Setter

Class FireSkill

Constructor

FireSkill(int x, int y,int direction);

Method

void update();

void draw();

boolean isVisible();

Class IceSkill

Constructor

IceSkill(int x, int y,int direction);

Method

void update();

void draw();

boolean isVisible();

Class MeteorSkill

Field

int frameCountX, frameCountY;

Constructor

MeteorSkill(int x, int y,int direction);

Method

void update();

void play();

void stop();

void draw();

boolean isVisible();

Class PoisonSkill

Constructor

PoisonSkill(int x, int y,int direction);

Method

void update();

void draw();

boolean isVisible();

Class SpikeSkill

Constructor

SpikeSkill(int x, int y,int direction);

Method

void update();

void draw();

boolean isVisible();

Package entity.monster

Class Cage

Field

Cage instance;

List<Monster> cage;

Method

Cage getInstance();

void add(String n, Land l, Hero h);

void remove(int i);

void updateAll();

void removeAll();

List<Monster> getCage();

Abstract Class Monster

Field

int x, y, hp, type, attack, level, direction, countWalk ,frameCountWalk, countDead, frameCountDead, countPanic, damageTaken, maxHp;

double xHp;

boolean isDead, isPanic;

Land land;

Hero hero;

Method

abstract void hit(Hero hero, Skill skill);

abstract void update();

void setX(int x);

Getter & Setter

Class Golem

Constructor

Golem(int x, int y, Land land, Hero hero)

Method

void update();

void deadAnimation();

void panicAnimation();

void walkAnimation();

void walkRight();

void walkLeft();

void draw(Graphics2D g)

void hit(Hero hero, Skill skill)

boolean isVisible();

int getZ();

Class Yeti

Field

int countAttack, frameCountAttack, attackDirection, speedDecrease;

boolean isAttack;

int[] debuff;

int[] countDebuff;

int[] timeCountDebuff ;

Constructor

Yeti(int x, int y, Land land, Hero hero)

Method

void update();

void deadAnimation();

void panicAnimation();

void walkAnimation();

void walkRight();

void walkLeft();

void debuffDamage();

void damageFromDebuff(Hero hero, int damage);

void draw(Graphics2D g)

void hit(Hero hero, Skill skill)

boolean isVisible();

int getZ();

Package ui

Class Banner

Field

boolean isVisible;

Method

void draw(Graphics2D g)

boolean isVisible();

boolean setVisible();

int getZ();

void run();

Class HeroStatus

Field

int score, health, mana, level,maxHp,maxMp,maxXp,xp;

double xHp,xMp,xXp;

boolean isDead;

Constructor

HeroStatus();

Method

void update();

void addXp(int x);

void setMaxXp();

void setMaxHp();

void setMaxMp();

void setCurrentHp(int x);

void setCurrentMp(int x);

void getScore;

void addScore(int score);

void resetLevel();

void getLevel();

void changeLevel()

void draw(Graphics2D g)

boolean isVisible();

int getZ();

Class SkillStatus

Field

double fAngle , iAngle, mAngle, pAngle, sAngle;

boolean fCasting, iCasting, mCasting, pCasting, sCasting;

boolean isPause;

Method

void draw();

boolean isVisible();

int getZ();

Getter & Setter