

Battle of Realm Document

Create by

Nonthakorn Taboonpong 6031029021

Kueakun Liwcharoenchai 6030057421

2110215 Programming Methodology

Semester 1 Year 2018

Chulalongkorn University

Battle of Realm

Dawn of The Holy Knight

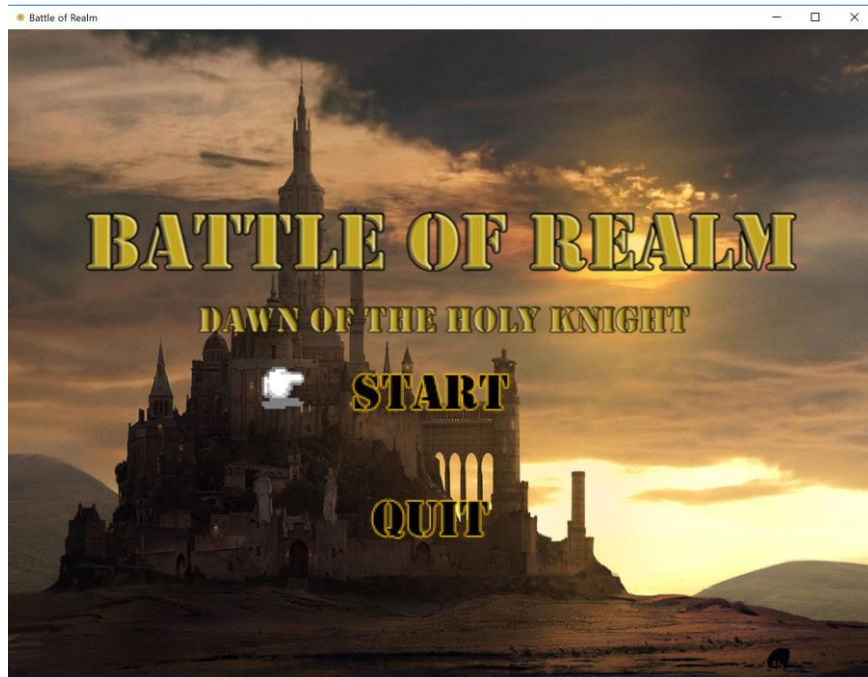
อาณาจักร Citeria อุดมสมบูรณ์ไปด้วยทรัพยากรต่าง ๆ ประชาชนมีชีวิตรุ่งเรืองด้วยความสงบสุข
ความเมตตา และความจงรักภักดีต่อกษัตริย์ Oblak ผู้ยิ่งใหญ่ วันหนึ่ง
อำนาจมืดจากโลกแห่งความตายเข้าคุกคามอาณาจักร จอมมาร Godin
ผู้ร้ายกาจและกองทัพของเขาได้บุกทำลายเมืองจนสิ้น กษัตริย์ Oblak ถูกลอบสังหาร
เมืองที่เคยอุดมสมบูรณ์กลับกลายเป็นเมืองที่ถูกปกคลุมด้วยมนตร์ดำ
ประชาชนที่เหลืออยู่ถูกปกครองด้วยความกลัว เป็นหน้าที่ของสามทหารเสือ Mill The Archer , Ramos The
Warrior และ Shai The Wizard ที่จะกอบกู้อาณาจักรจากมือของจอมมาร Godin
และนำพาความสงบสุขกลับคืนมาอีกครั้ง

Introduction

Battle of Realm is a RPG game. The goal of the game is to clear all monsters in every stage and
beat the boss, Godin. Every heroes have their own attributes and skills. You have to plan a fight in every
round carefully in case you want to win this game!

User Manual

Start Scene



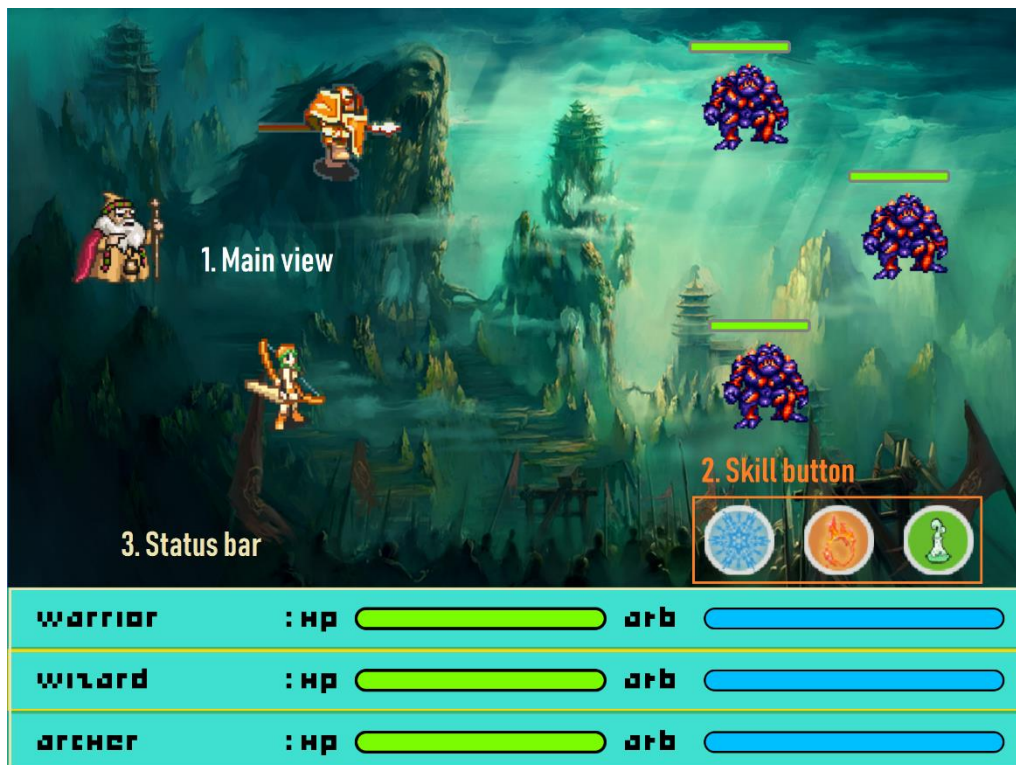
To start the game, click on start button. To quit the game, click on quit button.



These are all stages which you have to win. Click on next button to start the first stage.

Game Scene

The game scene consists of 3 major components.



1. Main view

Main view is the view that show everything in the current stage.

2. Skill button










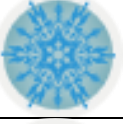


Skill button shows all skills of the hero in each round.

3. Status bar

Status bar shows the following list

- Type
- Current HP and Max HP
- Current ATB and Max ATB

Heroes and Skills

Heroes	Skills	
 Archer		Normal Attack
		Arrow Rain
		Fire Bow
 Warrior		Normal Attack
		Double Blade
		Eternal Flame
 Wizard		Frozen
		Burn
		Heal

Monsters

Monsters
 Bad Wizard



BlackDragon



Griffin

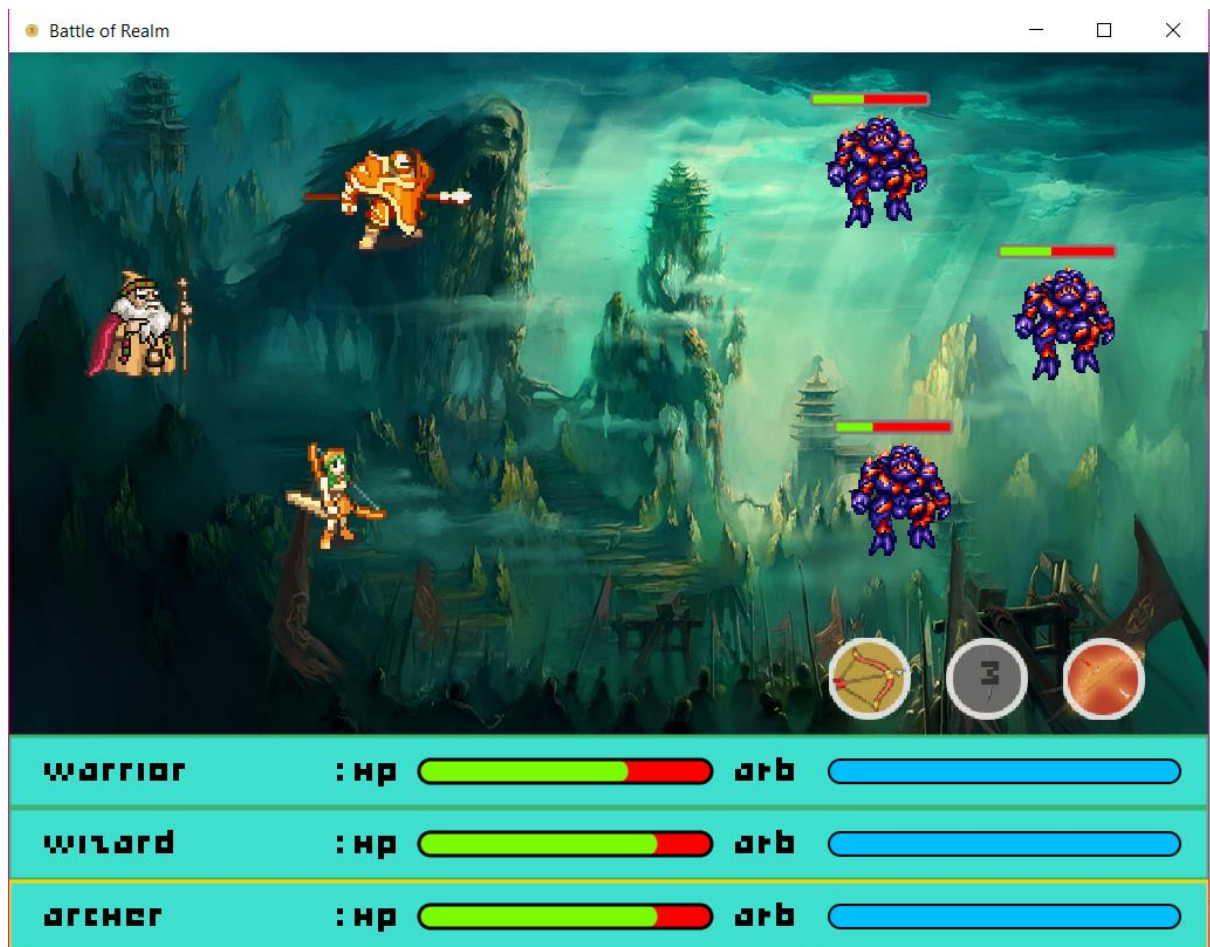


Lava man



Godin

Maps



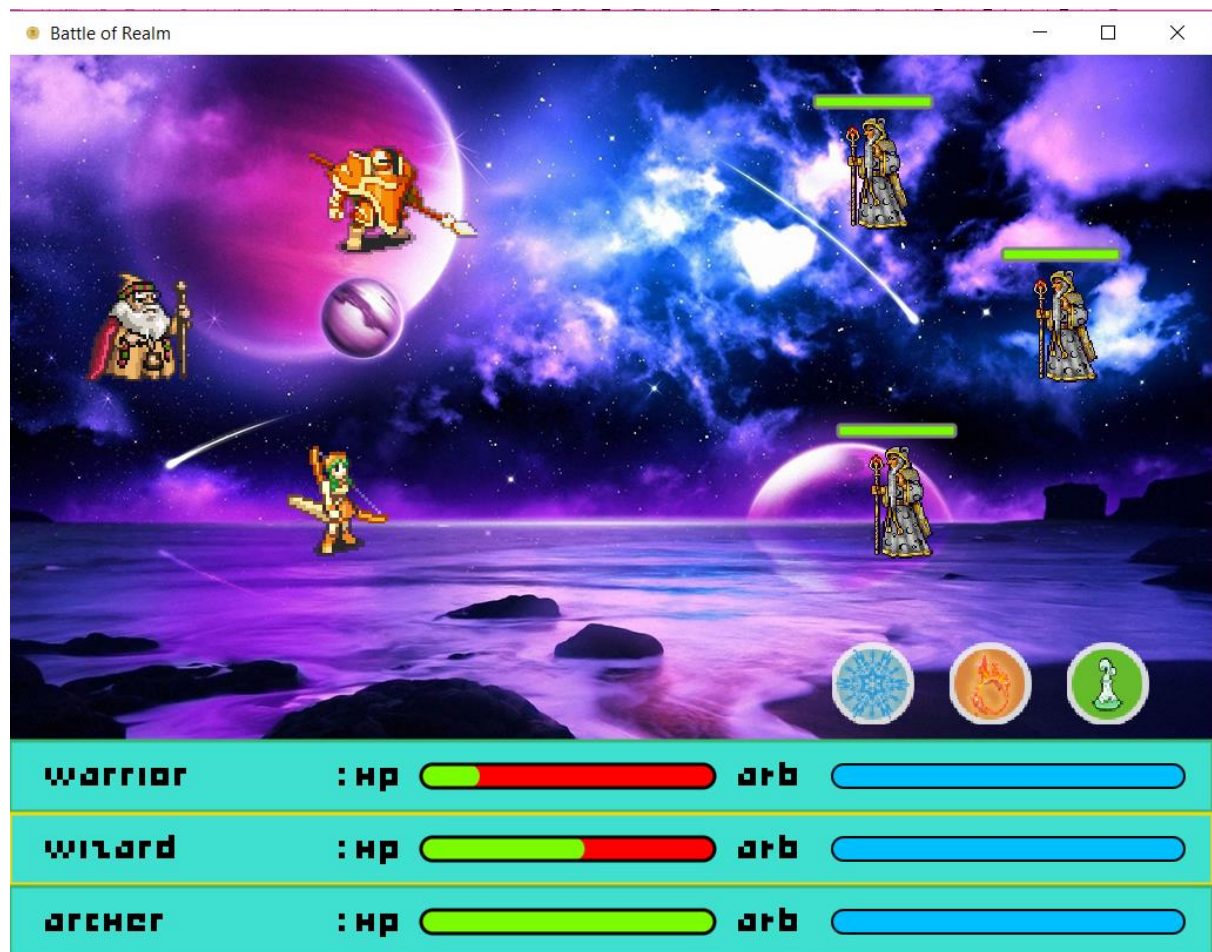
stage1



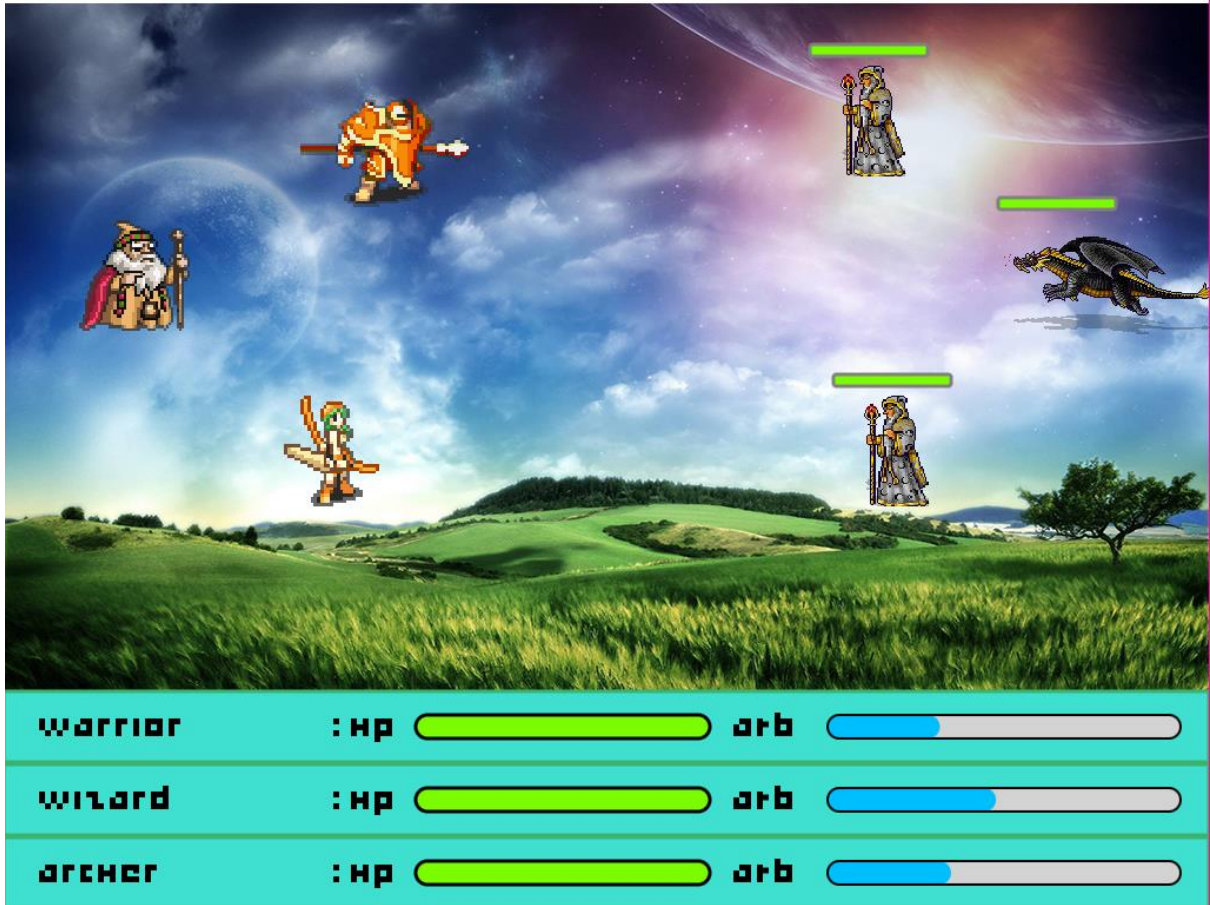
stage2



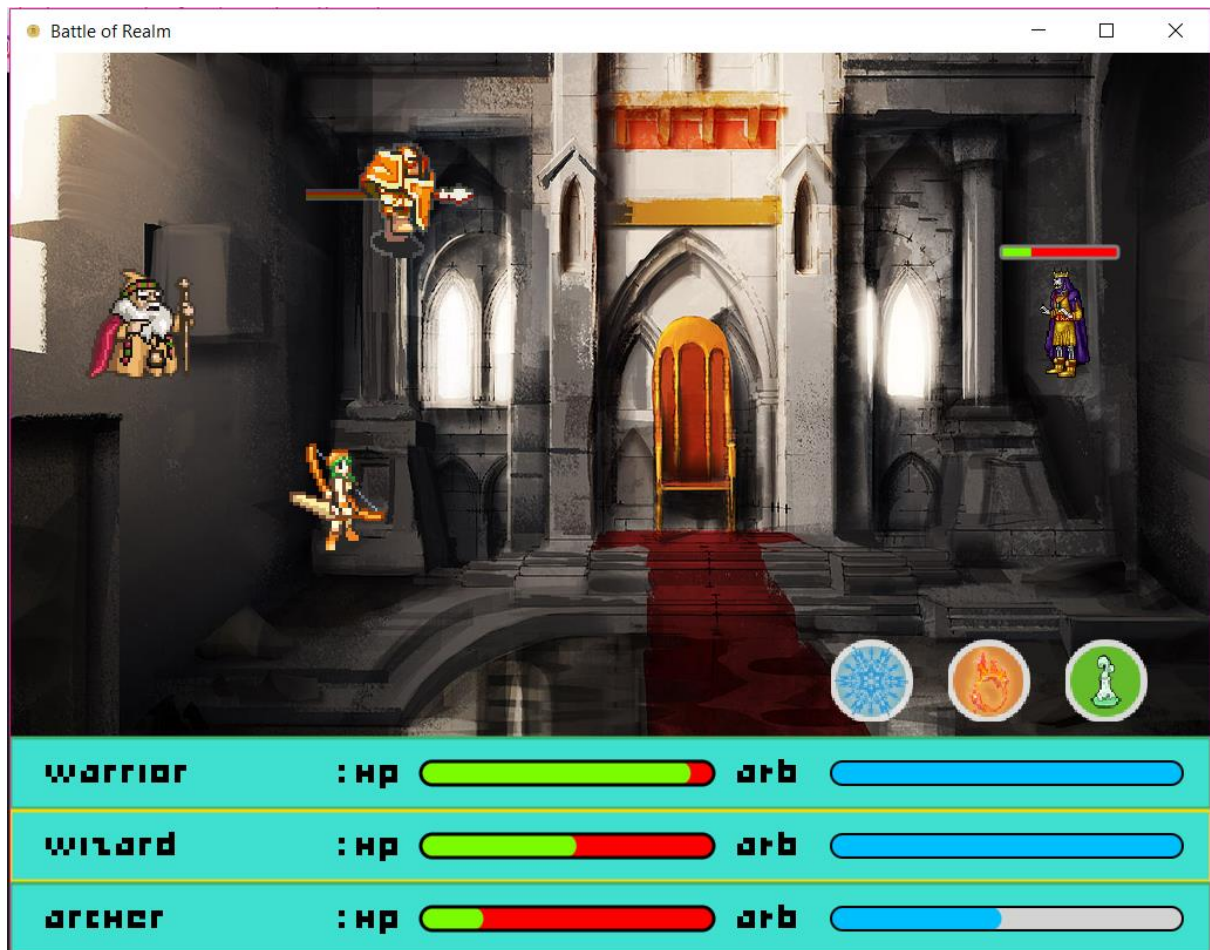
stage3



stage4

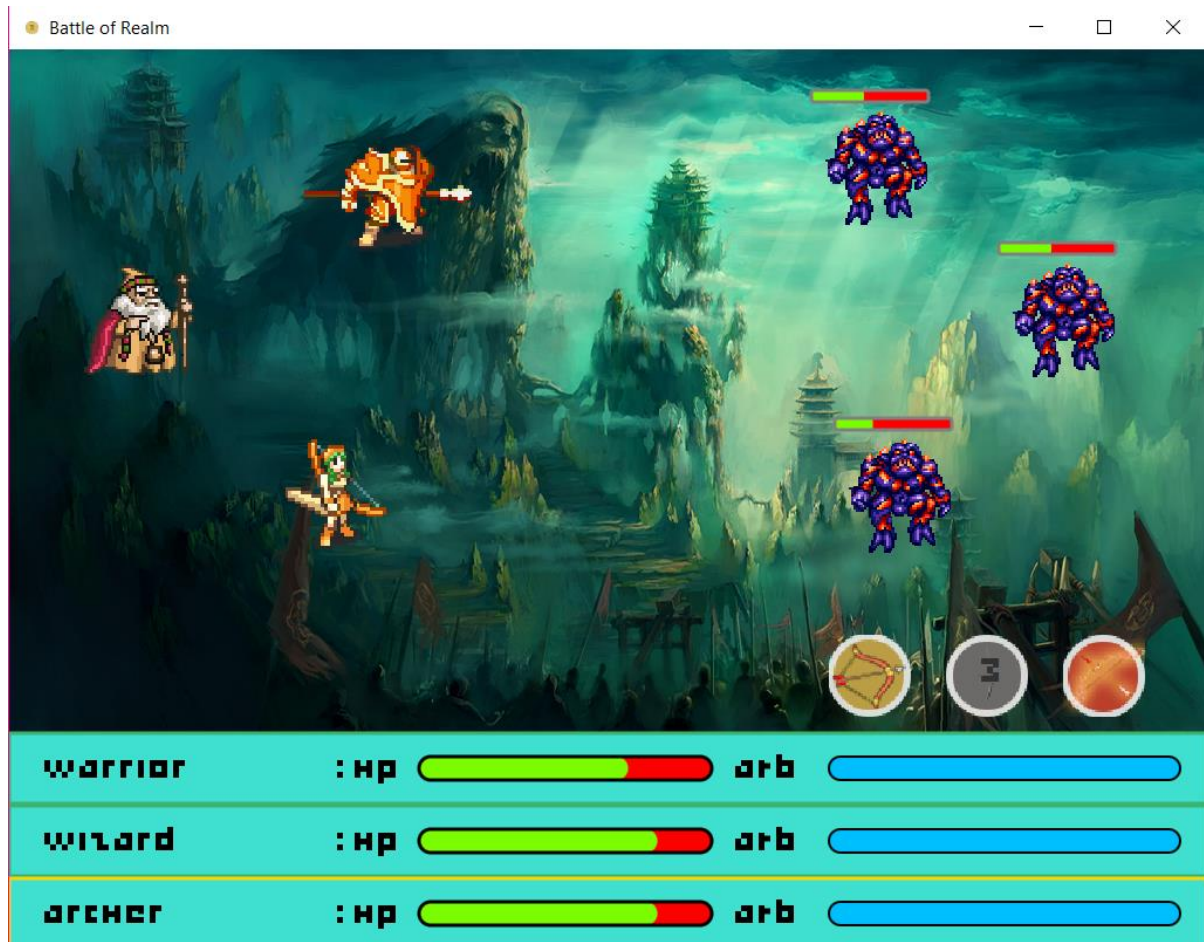


stage5

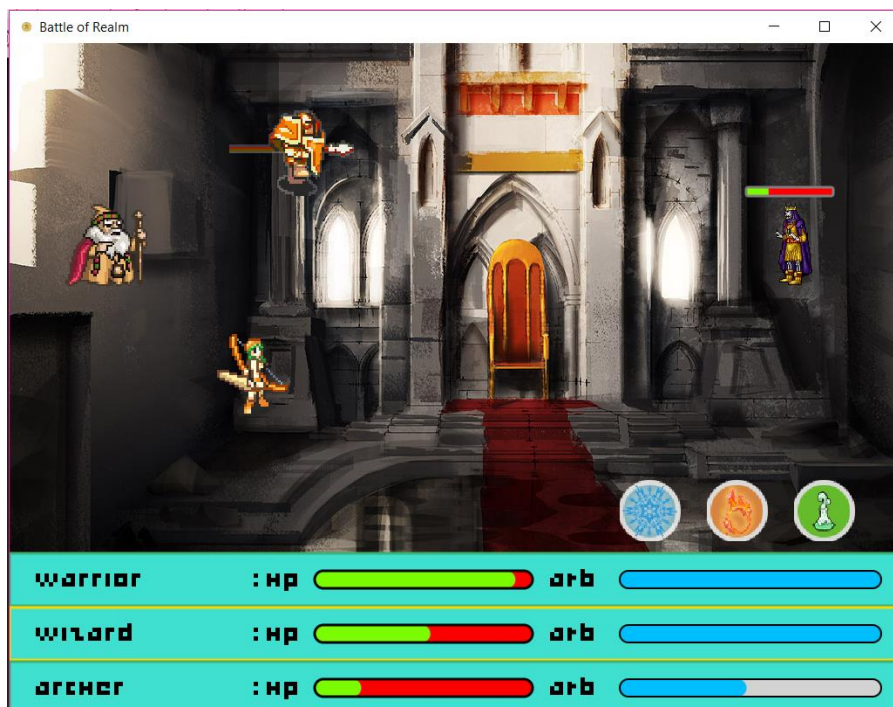
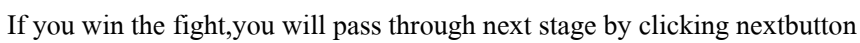


Boss Stage

Gameplay



This game is simple to play. There are queues between heroes and monsters. You can use one of 3 skills in your queue. Every skills of heroes are attack skills except one of wizard's skill, Heal, which use to increase HP of your team because your team will be attacked by monsters when the queue is on theirs. Every skills can use by clicking your mouse on skills then click on your target. However, some skills have their cooldown. So, it's your challenge to plan your fight carefully.



Remember, the more stage you pass the more difficulty you face!

Implementation Detail

1.1 Class Constant

1.1.1 Field

+ static final int SCREEN_WIDTH	Screen Width Constant.
+ static final int SCREEN_HEIGHT	Screen Height Constant.
+ static final int[][] warriorStat	Array of warrior statuses.
+ static final int[][] wizardStat	Array of wizard statuses.
+ static final int[][] archerStat	Array of archer statuses.

1.2 Class Fonts

1.2.1 Field

+ static final Font mainFont	Main font constant.
+ static final Font nameFont	Name font constant.
+ static final Font statFont	Status font constant.

1.3 Class Images

1.3.1 Field

+ static final Image cursor	Cursor Image.
+ static final Image nextButton	Next Button Image.
+ static final Image nextButtonHover	Next Button hover Image.
+ static final Image titleLogo	Title Logo Image.
+ static final Image[] worldMaps	Array of Worldmap Images.
+ static final Image stage1BG - + static final Image stage5BG	Stage 1-5 Background Images.
+ static final Image bossBG	Stage boss Image.
+ static final Image warriorIdle1 - + static final Image warriorIdle3	Warrior idle 1-3 Images.

+ static final Image warriorAtk1- + static final Image warriorAtk2	Warrior attack 1-2 Images.
+ static final Image warriorSkill1 - public static final Image warriorSkill5	Warrior skill 1-5 Images.
+ static final Image warriorUlti	Warrior ultimate Image.
+ static final Image wizardIdle1 - + static final Image wizardIdle3	Wizard idle 1-3 Images.
+ static final Image wizardAtk1 - + static final Image wizardAtk2	Wizard attack 1-2 Images.
+ static final Image archerIdle1 - + static final Image archerIdle3	Archer idle 1-3 Images.
+ static final Image archerAtk1 - + static final Image archerAtk3	Archer attack 1-3 Images.
+ static final Image archerSkill	Archer skill image.
+ static final Image archerUlti1 - + static final Image archerUlti4	Archer ulti image.
+ static final Image archerNormalIcon	Archer normal attack icon Image.
+ static final Image archerSkillIcon	Archer skill icon Image.
+ static final Image archerUltiIcon	Archer ulti icon Image.
+ static final Image archerNormalCD	Archer normal attack icon (cooling down) Image.
+ static final Image archerSkillCD	Archer skill icon (cooling down) Image.
+ static final Image archerUltiCD	Archer ulti icon (cooling down) Image.
+ static final Image warriorNormalIcon	Warrior normal attack icon Image.
+ static final Image warriorSkillIcon	Warrior skill icon Image.
+ static final Image warriorUltiIcon	Warrior ulti icon Image.
+ static final Image warriorNormalCD	Warrior normal attack icon (cooling down) Image.
+ static final Image warriorSkillCD	Warrior normal skill icon (cooling down) Image.
+ static final Image warriorUltiCD	Warrior normal ulti icon (cooling down) Image.
+ static final Image wizardSkill1Icon	Wizard skill1 icon Image.
+ static final Image wizardSkill2Icon	Wizard skill2 icon Image.

+ static final Image wizardHealIcon	Wizard heal icon Image.
+ static final Image wizardSkill1CD	Wizard skill1 icon(cooling down) Image.
+ static final Image wizardSkill2CD	Wizard skill2 icon(cooling down) Image.
+ static final Image wizardHealCD	Wizard heal icon(cooling down) Image.
+ static final Image lavaIdle1 - + static final Image lavaIdle2	Lava Man idle 1-2 Images.
+ static final Image lavaAtk1 – + static final Image lavaAtk3	Lava Man attack 1-3 Images.
+ static final Image griffinIdle1 - + static final Image griffinIdle2	Griffin idle 1-2 Images.
+ static final Image griffinAtk1 - + static final Image griffinAtk2	Griffin attack 1-2 Images.
+ static final Image badWizardIdle1 - + static final Image badWizardIdle3	Bad Wizard idle 1-3 Images.
+ static final Image badWizardAtk1 - + static final Image badWizardAtk2	Bad Wizard attack 1-2 Images.
+ static final Image dragonIdle1 - + static final Image dragonIdle2	Dragon idle 1-2 Images.
+ static final Image dragonAtk1 - + static final Image dragonAtk2	Dragon attack 1-2 Images.
+ static final Image bossIdle1 - + static final Image bossIdle3	Boss idle 1-3 Images.
+ static final Image bossAtk1 - + static final Image bossAtk3	Boss attack 1-3 Images.
+ static final ArrayList<Image> lavaIdles	List of Lava Man idles Image.
+ static final ArrayList<Image> lavaAtks	List of Lava Man attack Image.
+ static final ArrayList<Image> griffinIdles	List of Griffin idles Image.
+ static final ArrayList<Image> griffinAtks	List of Griffin attack Image.
+ static final ArrayList<Image> badWizardIdles	List of Bad Wizard idles Image.
+ static final ArrayList<Image> badWizardAtks	List of Bad Wizard attack Image.

+ static final ArrayList<Image> dragonIdles	List of Dragon idles Image.
+ static final ArrayList<Image> dragonAtks	List of Dragon attack Image.
+ static final ArrayList<Image> bossIdles	List of Boss idles Image.
+ static final ArrayList<Image> bossAtks	List of Boss attack Image.
+ static final Image normalParticle	Normal attack particle Image.
+ static final Image ultiParticle	Ulti particle Image.
+ static final Image wizard1Particle	Wizard's skill1 particle Image.
+ static final Image wizard2Particle	Wizard's skill2 particle Image.
+ static final Image wizardHealParticle	Wizard's heal particle Image.

1.3.2 Method

+ static void loadImages()	Method for load Images to arrays.
----------------------------	-----------------------------------

1.4 Class SFX

1.4.1 Fields

+ static final AudioClip startBGM	Start Background music track
+ static final AudioClip cursorMove	Cursor move Sound effect
+ static final AudioClip startSFX	Start Sound effect
+ static final AudioClip worldmapBGM	Worldmap Background music track

2.Package exception

2.1 Class CooldownException extends Exception

2.1.1 Field

- static final long serialVersionUID	SerialVersionUID for CooldownException
--------------------------------------	--

3.Package logic

3.1 Class GameManager

3.1.1 Field

- static GameManager instance	Get instance for GameManager
- boolean isMainScreen	Boolean for checking isMainScreen
- boolean isStageScreen	Boolean for checking isStageScreen
- boolean isGameScreen	Boolean for checking isGameScreen
- int stageNumber	Stage Number
- Warrior warrior	Warrior variable
- Wizard wizard	Wizard variable
- Archer archer	Archer variable
- StatusBar statusBar	StatusBar variable
- Map[] maps	Array containg all maps
- Queue<HealthEntity> turnQueue	Queue for calling turn
- ArrayList<HealthEntity> enemies	Enemies ArrayList
- ArrayList<HealthEntity> friends	Friends(player) ArrayList
- boolean isSelectSkill	Boolean for checking if the skill is selected
- boolean isSelectTarget	Boolean for checking if the target is selected
- int targetNo	Target Number

3.1.2 Constuctor

+ GameManager()	Constructor for GameManager Class.
-----------------	------------------------------------

3.1.3 Method

+ void addToQueue(HealthEntity e)	Add a HealthEntity object to turnQueue.
+ void nextQueue()	Call next queue from turnQueue.
+ HealthEntity getTopQueue()	Return top of the turnQueue.
+ boolean canCallQueue()	Return boolean if turnQueue is not empty.
+ void loadMapEntity()	Updates the RenderableHolder entities.
+ void update()	Update the status of the game, if friends are all dead, update the game into

	GameOverScreen. If enemies are all dead, update the game into LvUpScreen or VictoryScreen.
getter&setter	Getters and Setters of all fields.

3.2 Class LogicLoop

3.2.1 Field

- static LogicLoop instance	Get instance of LogicLoop.
- Timeline logicLoop	Timeline variable.
- boolean shouldRun	Boolean for checking if the loop should run.
- HealthEntity currentTurn	HealthEntity variable, contains the HealthEntity in this turn
- Warrior warrior	Variable for calling warrior from GameManager
- Wizard wizard	Variable for calling wizard from GameManager
- Archer archer	Variable for calling archer from GameManager

3.2.2 Constructor

+ LogicLoop()	Constructor for LogicLoop
---------------	---------------------------

3.2.3 Method

+ void update()	Calling the method RenderableHolder.getInstance().update(); updateATB(); callQueue();
+ void updateATB()	Update all of entities ATB gauge
+ void callQueue()	Call next queue.
Getter&setter	Getters and Setters for all fields.

3.3 Class MouseHandler

3.3.1 Field

- GameManager gm	Get instance for GameManager
- HealthEntity currentTurn	HealthEntity variable, contains the HealthEntity in this turn
- Skill currentSkill	Skill variable, contains the skill selected in this turn.

3.3.2 Constructor

+ MouseHandler(GameManager gm)	Constructor for MouseHandler
--------------------------------	------------------------------

3.3.3 Method

+ void handleClick(MouseEvent e)	Method for handle the target selecting.
+ int getTarget(int x, int y)	Return the target number
+ int[] getPos(int targetNo)	Return the target position (x,y)
+ boolean isInClickBox(int x1, int y1, int x2, int y2, int mouseX, int mouseY)	Check if mouse is click in the parameter

3.4 Class RenderLoop

3.4.1 Field

- static RenderLoop instance	Get instance for RenderLoop
- Timeline renderLoop	Timeline constant for renderLoop
- int renderTick	Int for counting the render tick

3.4.2 Constructor

+ RenderLoop()	Constructor for RenderLoop
----------------	----------------------------

3.4.3 Method

+ void render(GraphicsContext gc)	Render the elements in the RenderableHolder entities
getter	Getters for all fields

4.Package particle

4.1 Class NormalParticle implements IRenderable

4.1.1 Field

- int visibleTick	Int for counting the frames that particle can appear
- int maxVisibleTick	Int for limiting the maximum frames
- int x	Particle X position
- int y	Particle Y position

4.1.2 Constructor

+ NormalParticle(int x, int y)	Constructor for NormalParticle.
--------------------------------	---------------------------------

4.1.3 Method

+ boolean isVisible()	Return if particle is visible
+ boolean isExpired()	Return if particle is expired
+ boolean isDestroyed()	Return if particle is destroy
+ void draw(GraphicsContext gc)	Render the particle
getter	Getters for all fields

4.2 Class UtiParticle implements IRenderable

4.2.1 Field

- int visibleTick	Int for counting the frames that particle can appear
- int maxVisibleTick	Int for limiting the maximum frames
- int x	Particle X position
- int y	Particle Y position

4.2.2 Constructor

+ UtiParticle(int x, int y)	Constructor for UtiParticle
-----------------------------	-----------------------------

4.2.3 Method

+ boolean isVisible()	Return if particle is visible
+ boolean isExpired()	Return if particle is expired
+ boolean isDestroyed()	Return if particle is destroyed
+ void draw(GraphicsContext gc)	Render the particle
Getter	Getter for all fields

4.3 Class WizardHealParticle implements IRenderable

4.3.1 Field

- int visibleTick	Int for counting the frames that particle can appear
- int maxVisibleTick	Int for limiting the maximum frames
- int x	Particle X position
- int y	Particle Y position

4.3.2 Constructor

+ WizardHealParticle(int x, int y)	Constructor for WizardHealParticle.
------------------------------------	-------------------------------------

4.3.3 Method

+ boolean isVisible()	Return if particle is visible
+ boolean isExpired()	Return if particle is expired
+ boolean isDestroyed()	Return if particle is destroy
+ void draw(GraphicsContext gc)	Render the particle
Getter	Getter for all fields

4.4 Class WizardSkill1Particle implements IRenderable

4.4.1 Field

- int visibleTick	Int for counting the frames that particle can appear
- int maxVisibleTick	Int for limiting the maximum frames
- int x	Particle X position
- int y	Particle Y position

4.4.2 Constructor

+ WizardSkill1Particle(int x, int y)	Constructor for WizardSkill1Particle
--------------------------------------	--------------------------------------

4.4.3 Method

+ boolean isVisible()	Return if particle is visible
+ boolean isExpired()	Return if particle is expired
+ boolean isDestroyed()	Return if particle is destroy
+ void draw(GraphicsContext gc)	Render the particle
Getter	Getter for all fields

4.5 Class WizardSkill2Particle implements IRenderable

4.5.1 Field

- int visibleTick	Int for counting the frames that particle can appear
- int maxVisibleTick	Int for limiting the maximum frames
- int x	Particle X position
- int y	Particle Y position

4.5.2 Constructor

+ WizardSkill2Particle(int x, int y)	Constructor for WizardSkill2Particle
--------------------------------------	--------------------------------------

4.5.3 Method

+ boolean isVisible()	Return if particle is visible
+ boolean isExpired()	Return if particle is expired
+ boolean isDestroyed()	Return if particle is destroyed
+ void draw(GraphicsContext gc)	Render the particle
Getter	Getter for all fields

5. Package SharedObject

5.1 Interface IRenderable

5.1.1 Field

+ int getZ()	Interface for getZ() method
+ void draw(GraphicsContext gc)	Interface for draw() method
+ boolean isDestroyed()	Interface for isDestroyed() method
+ boolean isVisible()	Interface for isVisible() method

5.2 Class RenderableHolder

5.2.1 Field

- static final RenderableHolder instance	Get instance for RenderableHolder
- ArrayList<IRenderable> entities	List of all renderable entities
- Comparator<IRenderable> comparator	Comparator for sorting entities

5.2.2 Constructor

+ RenderableHolder()	Constructor for RenderableHolder
----------------------	----------------------------------

5.2.3 Method

+ void add(IRenderable entity)	Add entity to RenderableHolder entities list
+ void update()	Update the entities list if the entities is destroy
+ void clearEntities()	Clear the entities list

6.Package skill

6.1 Class *skill* (Abstract) implements IUsable

6.1.1 Field

- int maxCooldown	Maximum cooldown time int
- int currentCooldown	Current cooldown time int
- boolean isDisable	Boolean for checking if skill is disable
- boolean isSelected	Boolean for checking if skill is selected

- int targetNo	Target number int
----------------	-------------------

6.1.2 Constructor

+ Skill(int maxCooldown)	Constructor for Skill.
--------------------------	------------------------

6.1.3 Method

+ <i>String getName()</i>	Abstract method for returning name
+ <i>String getDescription()</i>	Abstract method for returning description
+ boolean isDisable()	Return if the skill is disable
+ boolean isReady()	Return if the skill is ready(Cooldown = 0)
+ boolean isSelected()	Return if the skill is selected
+ void reduceCooldown()	Reduce the cooldown by 1

6.2 Class HealSkill extends Skill

6.2.1 Field

- HealthEntity owner	HealthEntity variable contains the owner of the skill
----------------------	---

6.2.2 Constructor

+ HealSkill(int maxCooldown, HealthEntity owner)	Constructor for HealSkill.
--	----------------------------

6.2.3 Method

+ void use(HealthEntity target)	Use the skill to the target (Heal the target by 30%)
+ String getName()	Return name of the skill
+ String getDescription()	Return description of the skill

6.3 Class NormalAttack extends Skill

6.3.1 Field

- HealthEntity owner	HealthEntity variable contains the owner of the skill.
----------------------	--

6.3.2 Constructor

+ NormalAttack(int maxCooldown, HealthEntity owner)	Constructor for NormalAttack.
---	-------------------------------

6.3.3 Method

+ void use(HealthEntity target)	Use the skill to the target (Attacks the target)
+ String getName()	Return name of the skill
+ String getDescription()	Return description of the skill

6.4 Class SplashSkill extends Skill

6.4.1 Field

- HealthEntity owner	HealthEntity variable contains the owner of the skill.
----------------------	--

6.4.2 Constructor

+ SplashSkill(int maxCooldown, HealthEntity owner)	Constructor for SplashSkill.
--	------------------------------

6.4.3 Method

+ void useAll(ArrayList<HealthEntity> targets)	Attacks all of target in targets list
+ String getName()	Return name of the skill
+ String getDescription()	Return description of the skill
+ void use(HealthEntity target)	Use skill to the target (Attacks the target)

6.5 Class UltiSkill extends Skill

6.5.1 Field

- HealthEntity owner	HealthEntity variable contains the owner of the skill.
----------------------	--

6.5.2 Constructor

+ UltiSkill(int maxCooldown, HealthEntity owner)	Constructor for UltiSkill.
--	----------------------------

6.5.3 Method

+ String getName()	Return name of the skill
+ String getDescription()	Return description of the skill
+ void use(HealthEntity target)	Use skill to the target (Attacks the target with bonus attack damage)

6.6 Interface IUsable

+ void use(HealthEntity target)	Interface for use() method
+ boolean isSelected()	Interface for isSelected() method
+ String getName()	Interface for getName() method
+ String getDescription()	Interface for getDescription method

7.Package sprite

7.1 Class *Entity* (Abstract) implements IRenderable

7.1.1 Field

# double x	Double for x position
------------	-----------------------

# double y	Double for y position
# int z	Int for z position (Use in RenderableHolder)
# boolean visible	Boolean for checking if the Entity is visible
# boolean destroy	Boolean for checking if the Entity is destroyed
# int renderTick	Int for controlling the renderTick of the idle animation

7.1.2 Constructor

+ Entity(int z, double x, double y)	Constructor for Entity.
-------------------------------------	-------------------------

7.1.3 Method

+ boolean isVisible()	Return if entity is visible
+ boolean isDestroyed()	Return if entity is destroyy
getter&setter	Getter and Setter for all fields

7.2 Class *HealthEntity* (Abstract) extends Entity

7.2.1 Field

- int MAXHP	Max Hp int
-------------	------------

- int currentHP	Current Hp int
- int atk	Attack value int
- int defense	Defense value int
- int speed	Speed value int
- double criticalRate	Critical rate value int
- final double MAXATB	Max ATB(Active time battle) constant
- double currentATB	Curren ATB value int
- boolean inQueue	Boolean for check if the HealthEntity is in queue
- String name	String Name of the HealthEntity

7.2.2 Constructor

+ HealthEntity(int z, double x, double y,int maxHP, int atk, int def, int spd, double criRate, String name)	Constructor for HealthEntity
---	------------------------------

7.2.3 Method

+ <i>ArrayList<Image> getSkillIcon()</i>	Abstact method for getting skill icon
+ <i>ArrayList<Skill> getSkillList()</i>	Abstract method for getting skill list
+ <i>void setState(int state)</i>	Abstact method for set the state of the HealthEntity (Render state)

+ void update()	Update if the HealthEntity is dead, set the visible to false
+ boolean attack(HealthEntity e)	Attacks the target
+ void takeDamage(int damage)	Receive the damage with calculated damage
+ void heal(int hp)	Recovers the HealthEntity
+ boolean isDead()	Return if HealthEntity is dead (hp <= 0)
+ boolean isATBFull()	Return if the currentATB is full
+ boolean isInQueue()	Return if the HealthEntity is in queue
Getter&Setter	Getter and Setter for all fields.

8.Package sprite.enemy

8.1 Class Enemy

8.1.1 Field

- ArrayList<Image> idleImagesList	List of idle images
- ArrayList<Image> atkImagesList	List of attack images
- ArrayList<Skill> skillList	List of all skills
- int animationCounter	Int for counting the skill animation render tick
- int renderTick	Int for counting the idle animation render tick
- int state	Int for controlling state of enemy (Render state)
- double hpWidth	Double for controlling the enemy hp bar width

8.1.2 Constructor

+ Enemy(int z, double x, double y, int maxHP, int atk, int def, int spd, double criRate, String name, ArrayList<Image> idleImagesList, ArrayList<Image> atkImagesList)	Constructor for Enemy
--	-----------------------

8.1.3 Method

+ void updateSkillList()	Method for updating Skill lists
+ void draw(GraphicsContext gc)	Method for rendering the characters with different state
+ ArrayList<Image> getSkillIcon()	Return the skill icon images list
+ ArrayList<Skill> getSkillList()	Return the skills list
+ void setState(int state)	Set the render state

9.Package sprite.player

9.1 Class Archer extends HealthEntity

9.1.1 Field

- ArrayList<Image> idleImagesList	List of idle Images.
-----------------------------------	----------------------

- ArrayList<Image> atkImagesList	List of attack Images.
- ArrayList<Image> skillImagesList	List of skill Images.
- ArrayList<Image> ultiImagesList	List of ulti Images.
- ArrayList<Image> skillIconList	List of skill icon Images.
- ArrayList<Skill> skillList	List of skills.
- int state	Int for controlling state of character (Render state)
- int animationCounter	Int for controlling the animation render tick
- int renderTick	Int for controlling the renderTick of the idle animation

9.1.2 Constructor

+ Archer(int z, double x, double y, int maxHP, int atk, int def, int spd, double criRate, String name)	Constructor for Archer.
--	-------------------------

9.1.3 Method

+ void updateImageList()	Method for updating the Image lists
+ void updateSkillList()	Method for updating Skill lists
+ void draw(GraphicsContext gc)	Method for rendering the characters with different state
+ ArrayList<Image> getSkillIcon()	Return the skill icon images list

+ ArrayList<Skill> getSkillList()	Return the skills list
+ void setState(int state)	Set the render state
+ void updateStat(int[] stats)	Update the status of the character

9.2 Class Warrior extends HealthEntity

9.2.1 Field

- ArrayList<Image> idleImagesList	List of idle Images.
- ArrayList<Image> atkImagesList	List of attack Images.
- ArrayList<Image> skillImagesList	List of skill Images.
- ArrayList<Image> ultiImagesList	List of ulti Images.
- ArrayList<Image> skillIconList	List of skill icon Images.
- ArrayList<Skill> skillList	List of skills.
- int state	Int for controlling state of character (Render state)
- int animationCounter	Int for controlling the animation render tick
- int renderTick	Int for controlling the renderTick of the idle animation

9.2.2 Constructor

+ Warrior(int z, double x, double y, int maxHP, int atk, int def, int spd, double criRate, String name)	Constructor for Warrior
---	-------------------------

9.2.3 Method

+ void updateImageList()	Method for updating the Image lists
+ void updateSkillList()	Method for updating Skill lists
+ void draw(GraphicsContext gc)	Method for rendering the characters with different state
+ ArrayList<Image> getSkillIcon()	Return the skill icon images list
+ ArrayList<Skill> getSkillList()	Return the skills list
+ void setState(int state)	Set the render state
+ void updateStat(int[] stats)	Update the status of the character

9.3 Class Wizard extends HealthEntity

9.3.1 Field

- ArrayList<Image> idleImagesList	List of idle Images.
- ArrayList<Image> atkImagesList	List of attack Images.
- ArrayList<Image> skillImagesList	List of skill Images.
- ArrayList<Image> skillIconList	List of skill icon Images.
- ArrayList<Skill> skillList	List of skills.

- int state	Int for controlling state of character (Render state)
- int animationCounter	Int for controlling the animation render tick
- int renderTick	Int for controlling the renderTick of the idle animation

9.3.2 Constructor

+ Wizard(int z, double x, double y, int maxHP, int atk, int def, int spd, double criRate, String name)	Constructor for Wizard
--	------------------------

9.3.3 Method

+ void updateImageList()	Method for updating the Image lists
+ void updateSkillList()	Method for updating Skill lists
+ void draw(GraphicsContext gc)	Method for rendering the characters with different state
+ ArrayList<Image> getSkillIcon()	Return the skill icon images list
+ ArrayList<Skill> getSkillList()	Return the skills list
+ void setState(int state)	Set the render state
+ void updateStat(int[] stats)	Update the status of the character

10. Package UI

10.1 Class EndGameScreen extends Scene

10.1.1. Field

- Pane root	Pane variable
- static GraphicsContext gc	GraphicsContext variable

10.1.2. Constructor

+ EndGameScreen()	Constructor for EndGameScreen.
-------------------	--------------------------------

10.2 Class GameOverScreen extends Scene

10.2.1. Field

- Pane root	Pane variable
- static GraphicsContext gc	GraphicsContext variable

10.2.2. Constructor

+ GameOverScreen()	Constructor for GameOverScreen.
--------------------	---------------------------------

10.3 Class StageScreen extends Scene

10.3.1. Field

- Pane root	Pane variable
- static GraphicsContext gc	GraphicsContext variable

10.3.2. Constructor

+ StageScreen()	Constructor for StageScreen.
-----------------	------------------------------

10.3.3 Method

+ static void render()	Render the stage screen (render the world map)
------------------------	--

10.4 Class GameScreen extends Scene

10.4.1 Field

- Pane root	Pane variable
- static GraphicsContext gc	GraphicsContext variable
- GameManager gm	GameManager variable
- Map currentMap	Variable for controlling the current map
- HealthEntity currentTurn	Variable for controlling the currentTurn HealthEntity
- int skillNo	Variable for controlling which skill is chosen
- MouseHandler mouseHandler	Variable for MouseHandler object

- ImageView skill1	Skill 1 Button
- ImageView skill2	Skill 2 Button
- ImageView skill3	Skill 3 Button
- Label cooldown1	Skill 1 cooldown label
- Label cooldown2	Skill 2 cooldown label
- Label cooldown3	Skill 3 cooldown label

10.4.2. Constructor

+ GameScreen()	Constructor for GameScreen
----------------	----------------------------

10.4.3 Method

+ void mouseHandler(Canvas canvas)	Set the canvas action listener
+ void update()	Update the currentMap
+ void skillEventHandler(ImageView skill1, ImageView skill2, ImageView skill3)	EventListener for all skills button
+ void cooldownLabelHandler()	Label handler for all cooldown labels
Getter&Setter	Getter and Setter for all fields.

10.5 Class LvUpScreen extends Scene

10.5.1. Field

- Pane root	Pane variable
- static GraphicsContext gc	GraphicsContext variable
- GameManager gm	GameManager variavble

10.5.2. Constructor

+ LvUpScreen()	Constructor for LvUpScreen
----------------	----------------------------

10.5.3 Method

+ static void render()	Method for rendering the LvUpScreen
+ static void drawText()	Method for rendering all text in screen

10.6 Class mainScreen extends Scene

10.6.1. Field

- Pane root	Pane variable
- int mainScreenStage	Int for controlling mainScreen stage
- int cursorPosX	Int for controlling the cursor X position
- int cursorPosY[]	Array for controlling the cursor Y position

10.6.2. Constructor

+ mainScreen()	Constructor for MainScreen.
----------------	-----------------------------

10.6.3 Method

+ void buttonHandler(ImageView startButton, ImageView quitButton, GraphicsContext gc)	Method for setting the start and quit button EventListener
+ void render(GraphicsContext gc)	Render the screen background and the cursor

11. Package UI.map

11.1 Class *Map* (Abstract) implements IRenderable

11.1.1 Field

# Image mapImage	Map image
# int mapNumber	Map number
# boolean visible	Boolean for check if the map is visible.
# boolean destroy	Boolean for check if the map is destroyed

11.1.2 Constructor

+ Map(Image mapImage, AudioClip bgm, int mapNumber)	Constructor for Map
---	---------------------

11.1.3 Method

+ <i>ArrayList<HealthEntity> getEnemies()</i>	Abstact method for getEnemies() method
+ int getZ()	Return -9999
+ void draw(GraphicsContext gc)	Render the mapImage
+ boolean isDestroyed()	Return if the map is destroy
+ boolean isVisible()	Return if the map is visible
Getter&Setter	Getter and Setter for all fields

11.2 Class Map_Stage1 extends Map

11.2.1 Field

# ArrayList<HealthEntity> enemies	List of all enemies in the map
# Enemy e1	Enemies 1
# Enemy e2	Enemies 2
# Enemy e3	Enemies 3

11.2.2 Constructor

+ Map_Stage1()	Constructor for Map_Stage1
----------------	----------------------------

11.2.3 Method

+ ArrayList<HealthEntity> getEnemies()	Return List of all enemies
+ void draw(GraphicsContext gc)	Renders the enemies if the enemy is not dead

11.3 Class Map_Stage2 extends Map

11.3.1 Field

# ArrayList<HealthEntity> enemies	List of all enemies in the map
# Enemy e1	Enemies 1
# Enemy e2	Enemies 2
# Enemy e3	Enemies 3

11.3.2 Constructor

+ Map_Stage2()	Constructor for Map_Stage2
----------------	----------------------------

11.3.3 Method

+ ArrayList<HealthEntity> getEnemies()	Return List of all enemies
+ void draw(GraphicsContext gc)	Renders the enemies if the enemy is not dead

11.4 Class Map_Stage3 extends Map

11.4.1 Field

# ArrayList<HealthEntity> enemies	List of all enemies in the map
# Enemy e1	Enemies 1
# Enemy e2	Enemies 2
# Enemy e3	Enemies 3

11.4.2 Constructor

+ Map_Stage3()	Constructor for Map_Stage3
----------------	----------------------------

11.4.3 Method

+ ArrayList<HealthEntity> getEnemies()	Return List of all enemies
+ void draw(GraphicsContext gc)	Renders the enemies if the enemy is not dead

11.5 Class Map_Stage4 extends Map

11.5.1 Field

# ArrayList<HealthEntity> enemies	List of all enemies in the map
# Enemy e1	Enemies 1
# Enemy e2	Enemies 2
# Enemy e3	Enemies 3

11.5.2 Constructor

+ Map_Stage4()	Constructor for Map_Stage4
----------------	----------------------------

11.5.3 Method

+ ArrayList<HealthEntity> getEnemies()	Return List of all enemies
+ void draw(GraphicsContext gc)	Renders the enemies if the enemy is not dead

11.6 Class Map_Stage5 extends Map

11.6.1 Field

# ArrayList<HealthEntity> enemies	List of all enemies in the map
# Enemy e1	Enemies 1

# Enemy e2	Enemies 2
# Enemy e3	Enemies 3

11.6.2 Constructor

+ Map_Stage5()	Constructor for Map_Stage5
----------------	----------------------------

11.2.3 Method

+ ArrayList<HealthEntity> getEnemies()	Return List of all enemies
+ void draw(GraphicsContext gc)	Renders the enemies if the enemy is not dead

11.7 Class Map_StageBoss extends Map

11.7.1 Field

# ArrayList<HealthEntity> enemies	List of all enemies in the map
# Enemy e1	Enemies 1
# Enemy e2	Enemies 2
# Enemy e3	Enemies 3

11.7.2 Constructor

+ Map_StageBoss()	Constructor for Map_StageBoss
-------------------	-------------------------------

11.7.3 Method

+ ArrayList<HealthEntity> getEnemies()	Return List of all enemies
+ void draw(GraphicsContext gc)	Renders the enemies if the enemy is not dead

12.Package UI.statusbar

12.1 Class StatusBar implements IRenderable

12.1.1 Field

+ static final double BOX_HEIGHT	Box Height
+ static final double HP_ATB_BAR_HEIGHT	HP and ATB Bar Height
+ static final double HP_BAR_WIDTH	HP Bar Width
+ static final double ATB_BAR_WIDTH	ATB Bar Width
+ static final double HP_ATB_Y	HP and ATB Y-position
+ static final double HP_X	HP X-position
+ static final double ATB_X	ATB X-position

- static final Color BACKGROUND_COLOR	Background Colour
- static final Color HP_LEFT	HP left Colour
- static final Color HP_DEplete	HP deplete Colour
- static final Color ATB_COLOR	ATB Colour
- static final Color ATB_DEplete	ATB deplete Colour
- static final Color BORDER_COLOR	Border Colour
- static final Font NAME_FONT	Font of name
- static double warriorHpWidth	Warrior HP Width
- static double warriorAtbWidth	Warrior ATB Width
- static double wizardHpWidth	Wizard HP Width
- static double wizardAtbWidth	Wizard ATB Width
- static double archerHpWidth	Archer HP Width
- static double archerAtbWidth	Archer ATB Width
- static double warriorPosY	Warrior Y-position
- static double wizardPosY	Wizard Y-position
- static double archerPosY	Archer Y-position
# int z	z value
# boolean visible	boolean for check if status bar is visible
# boolean destroy	boolean for check if status bar is destroyed

12.1.2 Constructor

+ StatusBar(int z)	Constructor for StatusBar
--------------------	---------------------------

12.1.3 Method

+ void draw(GraphicsContext gc)	Renders the WarriorBox, WizardBox and ArcherBox
+ void drawWarriorBox(GraphicsContext gc)	Render the component in warrior status bar
+ void drawWizardBox(GraphicsContext gc)	Render the component in wizard status bar
+ void drawArcherBox(GraphicsContext gc)	Render the component in archer status bar
+ int getZ()	Return Z
+ boolean isDestroyed()	Return if the status bar is destroyed
+ boolean isVisible()	Return if the status bar is visible

13. Package utilities

13.1 Class AudioPlayer

13.1.1 Field

+ static AudioPlayer instance	Get instance for AudioPlayer
- MediaPlayer startBGM	StartBGM media player
- MediaPlayer worldmapBGM	WorldmapBGM media player

- MediaPlayer battleBGM	battleBGM media player
- MediaPlayer bossBGM	bossBGM media player
- MediaPlayer victoryBGM	victoryBGM media player
- MediaPlayer gameOverBGM	gameOverBGM media player
- MediaPlayer skill2SFX	Skill2SFC media player
- MediaPlayer healSFX	healSFX media player

3.1.2 Constructor

+ AudioPlayer()	Constructor for AudioPlayer
-----------------	-----------------------------

13.1.2 Method

+ void playCursorSound()	Play the cursor sound
+ void playStartSound()	Play the start sound
+ void playStartBGM()	Play the start background music
+ void pauseStartBGM()	Pause the start background music
+ void playWorldmapBGM()	Play the world map background music
+ void pauseWorldBGM()	Pause the world map background music
+ void playBattleBGM()	Play the battle background music

+ void pauseBattleBGM()	Pause the battle background music
+ void playBossBGM()	Play the boss background music
+ void pauseBossBGM()	Pause the boss background music
+ void playVictoryBGM()	Play the victory background music
+ void pauseVictoryBGM()	Pause the victory background music
+ void playGameOverBGM()	Play the game-over background music
+ void pauseGameOverBGM()	Pause the game-over background music
+ void playHitSound()	Play hit attack sound
+ void playUltiSound()	Play ulti sound
+ void playHealSound()	Play heal sound
+ void playSkill2Sound()	play skill2 sound
Getter	Getter for all fields.

13.2 Class Random

13.2.1 Method

+ static int randInt(int min, int max)	Random an integer (Inclusive).
--	--------------------------------

14.Package main

14.1 Class Main extends Application

14.1.1 Field

- static MainScreen mainScreen	MainScreen variable
- static StageScreen stageScreen	StageScreen variable
- static GameScreen gameScreen	GameScreen variable
- static LvUpScreen lvUpScreen	LvUpScreen variable
- static EndGameScreen endGameScreen	EndGameScreen variable
- static GameOverScreen gameOverScreen	GameOverScreen variable
- static Stage stage	Stage variable

14.1.2 Method

+ void start(Stage primaryStage)	Method for initialize the application
+ static void main(String[] args)	Main method
Getter	Getter for all fields.