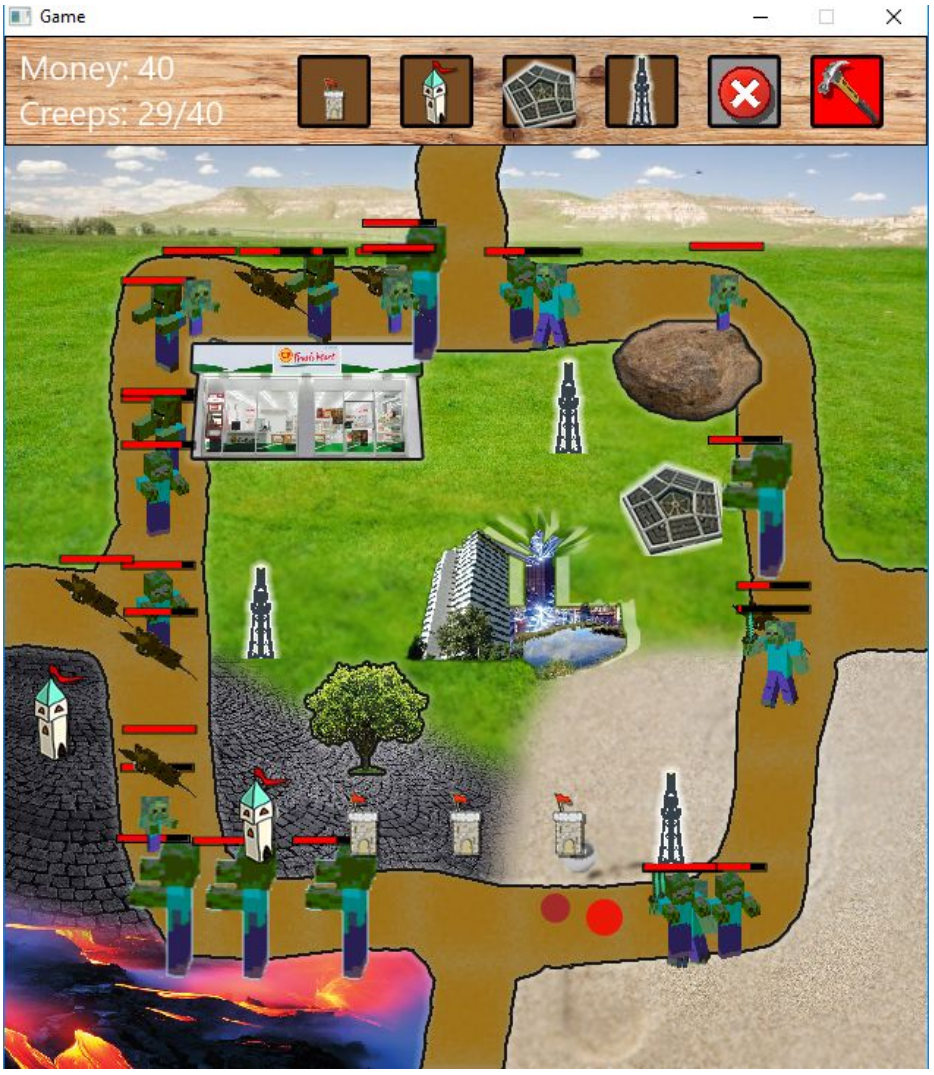
Small Square Tower Defense Game

Instruction

Tower Defense game is a game that the player has to kill the incoming creeps by placing the tower and the tower will attack a creep. In this game, creeps are moving in circle so the player have to keep the creep number to not exceed the maximum value or else the player will lose.



User Manual

There is a shopping tab for buying towers on the top of the screen. To view a tower information such as cost , range , damage and speed, hover your mouse over a tower icon. To buy and place a tower, you must have enough money and click on a tower icon to select a tower and place it on a panel that doesn't have a tower. The range of the selecting tower will be shown when hovering the mouse over a panel while selecting a tower.

3. Implementation Detail

UML

Figure x: The UML of the program

3.1 Package main

3.1.1 Class Main extends Application

3.1.1.1 Method

+ void start(Stage primaryStage)	The main entry point for the JavaFX applications. - Initialize movetarget. Initialize GameLogic, FieldDrawer, creepDrawer, BulletDrawer and TowerDrawer then add it to a pane. - Add pane to center of BorderPane. - Initialize menubar and add it to top of BorderPane. - Set scene to a scene that has BorderPane as Root
+ void main(String[] args)	An entry point of the application.

3.2 Package shopping

3.2.1 Class MouseValue

3.2.1.1 Field

+ int valueNow	A value that will change when holding something or not. Default is EMPTY.
+ int EMPTY	The mouse isn't holding anything.
+ int DESTROY	The mouse is holding destroy tool.
+ int NORMALTOWER	The mouse is holding normal tower.
+ int TALLTOWER	The mouse is holding tall tower.
+ int FASTTOWER	The mouse is holding fast tower.
+ int GODTOWER	The mouse is holding god tower.

3.2.2 Class Menubar extends Pane

3.2.2.1 Field

+ int money	The money that player has.

- Canvas display	A canvas that will draw money number and creep count

3.2.2.2 Constructor

+ Menubar()	Set PrefWidth to 630 and set PrefHeight to 75. Create five Rectangle and ToolTip for tower icon and set it to listen. Add background, display and Rectangles to this object.

3.2.2.3 Method

- void updateDisplay()	Update display. Draw money, creeps count and tower icon when money not enough.
+ void reduceMoney(int cost)	Reduce money by cost.
+ boolean checkMoneyLeft(int cost)	Called when build a tower. If money less than cost, set Mousevalue.ValueNow to EMPTY and return true , else return false.
+ void increaseMoney(int bounty)	Increase money by bounty.
- int creepsCount()	Return sum number of creep Count.

3.3 Package holder

3.3.1 Class Holder

3.3.1.1 Field

- Holder INSTANCE	The only object of this class.
- List<Creeps> creepHolder	This contain every creep currently in game.
- List<Tower> towerHolder	This contain every tower currently in game.
- List<Bullet> bulletHolder	This contain every bullet currently in game.
+ List<Image> creepsImage	This contain all of creeps image.
+ List<Image> towersImage	This contain all of towers image.
+ Image bulletImage	Bullet image.
+ Image bg	Background image.
+ Image end	Image that will show when game lose.
+ AudioClip clickedSound	Sound that will play when click a tower icon

	or a panel to build.
+ AudioClip shootSound	Sound that will play when a tower shoot.

3.3.1.2 Constructor

+ Holder()	Initialize all field.

3.3.1.3 Method

+ void loadResource()	Load all image and sound. This is called in static block.
+ Holder getInstance()	Return instance.
+ void addTower(Tower tower)	Add a tower to its holder.
+ void addCreep(Creeps Creep)	Add a creep to its holder.
+ void addBullet(Bullet bullet)	Add a bullet to its holder.
+ void update()	Check each creep and bullet if it's destroyed, remove from its holder.
+ getter of creepHolder, bulletHolder, towerHolder	Return the respective Holder.

3.4 Package drawer

3.4.1 Class FieldDrawer extends Canvas

3.4.1.1 Field

+ int NONETOWER	The state of a panel when it is not occupied by any tower.
+ int HAVETOWER	The state of a panel when it is occupied by tower.
+ int ROAD	The state of a panel that it is a way for creep walking.
+ int XSIZE	The number of rows of panels is 9.
+ int YSIZE	The number of column of panels is 9.
+ int PANELSIZE	The size of a panel.
+ int CANVASSIZE	The canvas is square so it's the width and height of canvas.

+ int[][] map	2D array represent the game field.

3.4.1.2 Constructor

+ FieldDrawer()	Initialize by calling superclass (canvas) constructor and draw background image.

3.4.1.3 Method

+ int getAt(int row, int column)	Get the state value of panel at row and column. If row or column is invalid, return -1.
+ int xToColumn(double x)	Return the column by x position.
+ int xToRow(double y)	Return the row by y position.
+ void changeAt(int row, int column, int newValue)	Change the state value of panel at row and column to newValue.

3.4.2 Class TowerDrawer extends Canvas

3.4.2.1 Constructor

+ TowerDrawer()	Call superclass constructor and add listener.

3.4.2.2 Method

+ void drawEnd()	fill canvas with black color and draw game end image.
+ void drawWin()	fill canvas with black color and draw game win image.
+ void addListener()	Set on mouse clicked for building tower. Set on mouse moved for showing tower range. Set on mouse exited for clear the screen. Redraw tower after every action.

3.4.3 Class CreepDrawer extends Canvas

3.4.3.1 Constructor

+ CreepDrawer()	Call superclass constructor.

3.4.3.2 Method

+ void drawCreeps()	Draw all creeps. Clear canvas before drawing.
+ void start()	Start adding creeps.
+ void addCreepDrawing(int k)	Add creeps in HowSummon by k creeps.

3.4.4 Class BulletDrawer extends Canvas

3.4.4.1 Constructor

+ CreepDrawer()	Call superclass constructor.

3.4.4.2 Method

+ void drawBullets()	Draw all Bullets. Clear canvas before drawing.

3.5 Package gamerule

3.5.1 Class HowSummon

3.5.1.1 Field

- Holder INSTANCE	The only object of this class.
+ int[] sleep	The pause between each wave.
+ int[] amount1	Amount of creeps to release in each wave.
+ List<Creeps> summonlist	List of creeps to summon.
- double westX	Configure West Position.
- double westY	Configure West Position.
- MoveTarget westTarget	Configure West Target.
- double southX	Configure South Position.
- double southY	Configure South Position.
- MoveTarget southTarget	Configure South Target.
- double eastX	Configure East Position.
- double eastY	Configure East Position.
- MoveTarget eastTarget	Configure East Target.
- double northX	Configure North Position.
- double northY	Configure North Position.
- MoveTarget northTarget	Configure North Target.

3.5.1.2 Constructor

+ HowSummon()	Initialize summonlist

3.5.1.3 Method

+ void load1()	Add creeps to summonlist.
- Creeps getCreep(double x, double y, MoveTarget target)	Return a new creep at position x,y and set it movetarget to target.
- RookieC getRookieCreep(double x, double y, MoveTarget target)	Return a new Rookie creep at position x,y and set it movetarget to target.
- StrongC getStrongCreep(double x, double y, MoveTarget target)	Return a new Strong creep at position x,y and set it movetarget to target.
- JiantC getJiantCreep(double x, double y, MoveTarget target)	Return a new Jiant creep at position x,y and set it movetarget to target.
- FighterC getFighterCreep(double x, double y, MoveTarget target)	Return a new Fighter creep at position x,y and set it movetarget to target.
- FlyAbleC getFlyAbleCreep(double x, double y, MoveTarget target)	Return a new FlyAble creep at position x,y and set it movetarget to target.
+ HowSummon getInstance()	Return instance.
+ List<Creeps> getSummonlist()	Return summonlist.

3.6 Package logic

3.6.1 Class GameLogic

3.6.1.1 Field

+ List<Creeps> creepsContainer	Another container from holder. This is for CreepDrawer to look and draw creeps.
+ int SLEEPTIME	Sleep time for each update loop and tower cooldown.
+ int MAXCREEP	Maximum number of creeps. If number of creeps more than this, the player lose.
+ boolean end	Flag indicate when the game end.
+ boolean win	Flag indicate when the game win.

3.6.1.2 Constructor

+ GameLogic()	- Do nothing.

3.6.1.3 Method

+ void start(FieldDrawer fieldDrawer,CreepDrawer creepDrawer,BulletDrawer bulletDrawer,TowerDrawer towerDrawer)	Initialize end = false. call updateLogic.
- void updateLogic(FieldDrawer fieldDrawer,CreepDrawer creepDrawer,BulletDrawer bulletDrawer,TowerDrawer towerDrawer)	Create a Thread that do - start creepDrawer at start - Loop make each tower shoot, drawers draw creep and bullet, check every creep if it is destroyed make it give money and remove from container, update creeps tower and bullet.

3.7 Package entity

3.7.1 Interface Entity

3.7.1.1 Method

+ boolean collideWith(AliveEntity other)	- Check if this collide with another AliveEntity.If distance between two object is more than sum of two radius, they're collided.
+ boolean isDestroyed()	- Return True if this object will be destroyed.
+ void move()	- Move this object.
+ void update();	- Update this object.

3.7.2 Abstract Class AliveEntity implements Entity

3.7.2.1 Field

+ double x	The x position.
+ double y	The y position.
+ double speed	The speed of this object.
# int radius	Radius of this object that will collide with other. If distance between object is more than sum of

	two radius.
# boolean isDestroyed	True if this object will be destroyed.

3.7.2.2 Method

+ boolean collideWith(AliveEntity other)	Check if this collide with another AliveEntity.If distance between two object is more than sum of two radius, they're collided.
+ boolean isDestroyed()	Return True if this object will be destroyed.
+ abstract void move()	- Move this object.
+ abstract void update()	- Update this object.

3.8 Package Bullet

3.8.1 Class Bullet extends AliveEntity

3.8.1.1 Field

+ Creep target	The creep target of this bullet
- Tower tower	The tower that shoot this bullet
- Image img	Image of bullet.
+ boolean boom	Status of Pre-destroyed.
- boolean miss	Status of miss target.

3.8.1.2 Constructor

+ Bullet(double x, double y, Tower tower)	Initialize all field.

3.8.1.3 Method

+ void update()	Update bullet status. - called findTarget() - move()
+ Creeps findTarget()	Find nearest creep (From this bullet)
+ void move()	Move x and y toward target.
- double distanceSize()	Calculate the distance between this bullet and the target.
+ void draw(GraphicsContext gc)	Draw image of this bullet at x,y

3.9 Package creeps

3.9.1 Class MoveTarget extends Point2D

3.1.1.1 Field

+ MoveTarget UPLEFT	The upper left target of map for creep moving.
+ MoveTarget UPRIGHT	The upper right target of map for creep moving.
+ MoveTarget DOWNLEFT	The down left target of map for creep moving.
+ MoveTarget DOWNRIGHT	The down right target of map for creep moving.
+ MoveTarget LEFT	The left target of map for creep moving.
+ MoveTarget DOWN	The down target of map for creep moving.
+ MoveTarget RIGHT	The right target of map for creep moving.
+ MoveTarget UP	The up target of map for creep moving.
+ MoveTarget NextTarget	The next target to move to after this object target.

3.9.1.2 Constructor

+ Bullet(double x, double y)	Initialize by calling superclass. (This doesn't initialize next target so it has to be manually set.)
+ Bullet(double x, double y, MoveTarget next)	Initialize by calling superclass and set the next target..

3.9.1.3 Method

+ void initial()	Initialize all static field.
+ void setNextTarget	Setter of NextTarget.

3.9.2 Class Creeps extends AliveEntity

3.9.2.1 Field

# int maxHp	Maximum and start hit point.
# int hp	The current hp.

# int bounty	The money that will be rewarded when this creep die.
# MoveTarget target	Point target to move to of this creep.
+ boolean boom	
+ Image img	Image of this creep.

3.9.2.2 Constructor

+ Bullet(double x, double y, Tower tower)	Initialize all field.

3.9.2.3 Method

+ void update()	Update status of creep. - If creep hp <= 0, change *isDestroyed* to true - else call move()
+ void move()	Move x and y toward target. If distance <= speed, change target to next target.
+ void tookDamage(int damage)	Reduce hp by damage
+ void draw(GraphicsContext gc)	Draw image of this bullet at x,y
+ void boom(GraphicsContext gc)	Status of Pre-destroyed.
+ void draw(GraphicsContext gc)	Draw image of this bullet at x,y

3.9.3 Class FighterC extends Creeps

3.9.3.1 Constructor

+ FighterC(double x, double y , MoveTarget target)	Initialize all field.

3.9.4 Class FlyAbleC extends Creeps

3.9.4.1 Constructor

+ FlyAbleC(double x, double y , MoveTarget target)	Initialize all field.

3.9.5 Class JiantC extends Creeps

3.9.5.1 Constructor

+ JiantC(double x, double y , MoveTarget target)	Initialize all field.

3.9.6 Class RookieC extends Creeps

3.9.6.1 Constructor

+ RookieC(double x, double y , MoveTarget target)	Initialize all field.

3.9.7 Class StrongC extends Creeps

3.9.7.1 Constructor

+ StrongC(double x, double y , MoveTarget target)	Initialize all field.

3.10 Package tower

3.10.1 Class Tower

3.10.1.1 Field

+ double x, y	Position in X , Y axis. (It is center of a panel)
+ int attack	This attack point.
+ int cooldown	Max cooldown
+ int cooldowncount	Current cooldown
+ boolean ready	Status of ready to shoot a bullet
+ Image img	Tower's image
+ int cost	Tower's cost
+ int range	Tower's shooting range
+ int COST	This cost
+ int RANGE	This range

3.10.1.2 Constructor

+ Tower(int row, int column)	Initialize all field.

3.10.1.3 Method

+ void update()	Update status of tower. - If creep dead, change *isDestroyed* to true
+ void shootBulletIfReady()	Add a bullet to draw and shoot if ready.
+ void move()	Move x and y toward target.
- double distanceTo(Creeps creep)	Calculate the distance between this tower and the creep.
+ void draw(GraphicsContext gc)	Draw image of this tower at x,y

3.10.2 Class FastT extends Tower

3.10.2.1 Constructor

+ FastT(int row, int column)	Initialize all field.

3.10.1.1 Field

+ int COST	This cost
+ int RANGE	This range

3.10.3 Class GodT extends Tower

3.10.3.1 Constructor

+ GodT(int row, int column)	Initialize all field.

3.10.3.1 Field

+ int COST	This cost
+ int RANGE	This range

3.10.4 Class TallT extends Tower

3.10.4.1 Constructor

+ TallT(int row, int column)	Initialize all field.

3.10.4.1 Field

+ int COST	This cost
+ int RANGE	This range

3.10.5 Class BigT extends Tower

3.10.5.1 Constructor

+ BigT(int row, int column)	Initialize all field.

3.10.5.1 Field

+ int COST	This cost
+ int RANGE	This range

3.11 Package impossibruException

3.11.5 Class ImpossibruException extends Exception

3.11.5.1 Constructor

+ ImpossibruException()	This exception is to be thrown where the code shouldn't have reached.