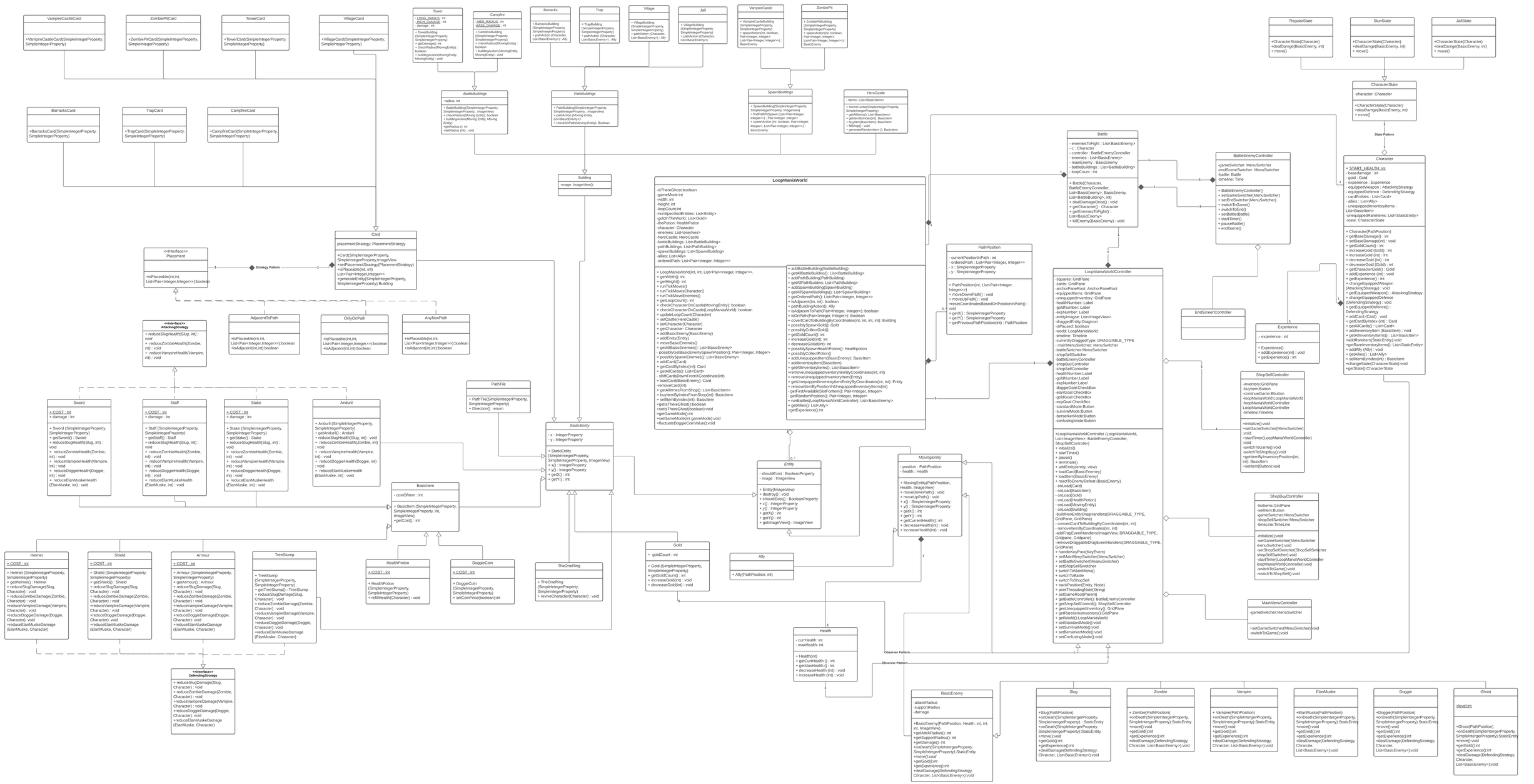
VampireCastleCard
+VampireCastleCard(SimpleIntegerProperty, SimpleIntegerProperty)
ZombiePitCard
+ZombiePitCard(SimpleIntegerProperty, SimpleIntegerProperty)
TowerCard
+TowerCard(SimpleIntegerProperty, SimpleIntegerProperty)
VillageCard
+VillageCard(SimpleIntegerProperty, SimpleIntegerProperty)
Tower
- LONG_RADIUS : int
HIGH_DAMAGE : int
- damage : int
+ TowerBuilding (SimpleIntegerProperty, SimpleIntegerProperty)
+ getDamage() : int
+ getRadius(MovingEntity) : boolean
+ buildingAction(MovingEntity, MovingEntity) : void
Campfire
- MED_RADIUS : int
BASE_DAMAGE : int
+ CampfireBuilding (SimpleIntegerProperty, SimpleIntegerProperty)
+ checkRadius(MovingEntity) : boolean
+ buildingAction (MovingEntity, MovingEntity) : void
Barracks
+ BarracksBuilding (SimpleIntegerProperty, SimpleIntegerProperty)
+ pathAction (Character, List<BasicEnemy>) : Ally
Trap
+ TrapBuilding (SimpleIntegerProperty, SimpleIntegerProperty)
+ pathAction (Character, List<BasicEnemy>) : Ally
Village
+ VillageBuilding (SimpleIntegerProperty, SimpleIntegerProperty)
+ pathAction (Character, List<BasicEnemy>) : Ally
Jail
+ VillageBuilding (SimpleIntegerProperty, SimpleIntegerProperty)
+ pathAction (Character, List<BasicEnemy>)
VampireCastle
+ VampireCastleBuilding (SimpleIntegerProperty, SimpleIntegerProperty)
+ spawnAction(int, boolean, Pair<Integer, Integer>, List<Pair<Integer, Integer>>) : BasicEnemy
ZombiePit
+ ZombiePitBuilding (SimpleIntegerProperty, SimpleIntegerProperty)
+ spawnAction(int, boolean, Pair<Integer, Integer>, List<Pair<Integer, Integer>>) : BasicEnemy
BarracksCard
+BarracksCard(SimpleIntegerProperty, SimpleIntegerProperty)
TrapCard
+TrapCard(SimpleIntegerProperty, SimpleIntegerProperty)
CampfireCard
+CampfireCard(SimpleIntegerProperty, SimpleIntegerProperty)
BattleBuildings
-radius: int
+ BattleBuilding(SimpleIntegerProperty, SimpleIntegerProperty , ImageView)
+ checkRadius(Moving Entity): boolean
+ buildingAction(Moving Entity, Moving Entity)
+getRadius () : int
+setRadius (int) : void
PathBuildings
+ PathBuilding(SimpleIntegerProperty, SimpleIntegerProperty , ImageView)
+ pathAction (Moving Entity, List<BasicEnemy>)
+ checkOnPath(Moving Entity): Boolean
SpawnBuildings
+ SpawnBuilding(SimpleIntegerProperty, SimpleIntegerProperty, ImageView)
+ findPathToSpawn (List<Pair<Integer, Integer>> , Pair<Integer, Integer>)
+ spawnAction (int, boolean, Pair<Integer, Integer>, List<Pair<Integer, Integer>>) : BasicEnemy
HeroCastle
- items: List<BasicItem>
+ HeroCastle(SimpleIntegerProperty, SimpleIntegerProperty)
+ getAllItems() : List<BasicItem>
+ getItemByIndex(int): BasicItem
+ buyItem(BasicItem): BasicItem
+ lifeUp() : void
+ generateRandomItem () : BasicItem
Building
-image: ImageView()
Card
placementStrategy: PlacementStrategy
+Card(SimpleIntegerProperty, SimpleIntegerProperty,ImageView
+setPlacementStrategy(PlacementStrategy)
+isPlaceable(int, int) List<Pair<Integer,Integer>>
+generateEntity(SimpleIntegerProperty, SimpleIntegerProperty):Building
<<Interface>>
Placement
+isPlaceable(int,int, List<Pair<Integer,Integer>>):boolean
Strategy Pattern
<<Interface>>
AttackingStrategy
+ reduceSlugHealth(Slug, int) : void
+ reduceZombieHealth(Zombie, int) : void
+ reduceVampireHealth(Vampire, int) : void
AdjacentToPath
+isPlaceable(int,int, List<Pair<Integer,Integer>>):boolean
+isAdjacent(int,int):boolean
OnlyOnPath
+isPlaceable(int,int, List<Pair<Integer,Integer>>):boolean
+isAdjacent(int,int):boolean
AnyNonPath
+isPlaceable(int,int, List<Pair<Integer,Integer>>):boolean
+isAdjacent(int,int):boolean
Sword
+ COST : int
+ damage : int
+ Sword (SimpleIntegerProperty, SimpleIntegerProperty)
+ getSword() : Sword
+ reduceSlugHealth(Slug, int) : void
+ reduceZombieHealth(Zombie, int) : void
+ reduceVampireHealth(Vampire, int) : void
+ reduceDoggieHealth(Doggie, int) : void
+ reduceElanMuskeHealth (ElanMuske, int) : void
Staff
+ COST : int
+ damage : int
+ Staff (SimpleIntegerProperty, SimpleIntegerProperty)
+ getStaff() : Staff
+ reduceSlugHealth(Slug, int) : void
+ reduceZombieHealth(Zombie, int) : void
+ reduceVampireHealth(Vampire, int) : void
+ reduceDoggieHealth(Doggie, int) : void
+ reduceElanMuskeHealth (ElanMuske, int) : void
Stake
+ COST : int
+ damage : int
+ Stake (SimpleIntegerProperty, SimpleIntegerProperty)
+ getStake() : Stake
+ reduceSlugHealth(Slug, int) : void
+ reduceZombieHealth(Zombie, int) : void
+ reduceVampireHealth(Vampire, int) : void
+ reduceDoggieHealth(Doggie, int) : void
+ reduceElanMuskeHealth (ElanMuske, int) : void
Anduril
+ Anduril (SimpleIntegerProperty, SimpleIntegerProperty)
+ getAnduril() : Anduril
+ reduceSlugHealth(Slug, int) : void
+ reduceZombieHealth(Zombie, int) : void
+ reduceVampireHealth(Vampire, int) : void
+ reduceDoggieHealth(Doggie, int) : void
+ reduceElanMuskeHealth (ElanMuske, int) : void
PathTile
+ PathTile(SimpleIntegerProperty, SimpleIntegerProperty)
+ Direction() : enum
LoopManiaWorld
-isThereGhost:boolean
-gameMode:int
-width: int
-height: int
-loopCount:int
-nonSpecifiedEntities: List<Entity>
-goldinTheWorld: List<Gold>
-thePotion: HealthPotion
-character: Character
-enemies: List<enemies>
-heroCastle: HeroCastle
-battleBuildings: List<BattleBuilding>
-pathBuildings: List<PathBuilding>
-spawnBuildings: List<SpawnBuilding>
-allies: List<Ally>
-orderedPath: List<Pair<Integer, Integer>>
+ LoopManiaWorld(int, int, List<Pair<Integer, Integer>>.
+ getWidth() : int
+ getHeight() : int
+ runTickMoves()
+ runTickMovesCharacter()
+ runTickMoveEnemies()
+ getLoopCount() : int
+ checkCharacterOnCastle(MovingEntity): boolean
+ checkCharacterOnCastle(LoopManiaWorld): boolean
+ updateLoopCount(Character)
+ setCastle(HeroCastle)
+ setCharacter(Character)
+ getCharacter: Character
+ addBasicEnemy(BasicEnemy)
+ addEntity(Entity)
+ moveBasicEnemies()
+ getAllBasicEnemies() : List<BasicEnemy>
- possiblyGetBasicEnemySpawnPosition(): Pair<Integer, Integer>
+ possiblySpawnEnemies() : List<BasicEnemy>
+ addCard(Card)
+ getCardByIndex(int): Card
+ getAllCards() : List<Card>
- shiftCardsDownFromXCoordinate(int)
+ loadCard(BasicEnemy): Card
-removeCard(int)
+ getAllItmesFromShop() : List<BasicItem>
+ buyItemByIndexFromShop(int): BasicItem
+ sellItemByIndex(int): BasicItem
+getIsThereGhost():boolean
+setIsThereGhost(boolean):void
+getGameMode():int
+setGameMode(int gameMode):void
+fluctuateDoggieCoinValue():void
+ addBattleBuilding(BattleBuilding)
+ getAllBattleBuildings() : List<BattleBuilding>
+ addPathBuilding(PathBuilding)
+ getAllPathBuildings: List<PathBuilding>
+ addSpawnBuilding(SpawnBuilding)
+ getAllSpawnBuildings() : List<SpawnBuilding>
+ getOrderedPath() : List<Pair<Integer, Integer>>
+ isAdjacent(int, int): boolean
+ pathBuildingAction(): Ally
+ isAdjacentToPath(Pair<Integer, Integer>): boolean
+ isOnPath(Pair<Integer, Integer>): Boolean
+ coverCardToBuildingByCoordinates(int, int, int, int): Building
+ possiblySpawnGold(): Gold
+ possiblyCollectGold()
+ getGoldCount(): int
+ increaseGold(int): int
+ decreaseGold(int): int
+ possiblySpawnHealthPotion(): HealthPotion
+ possiblyCollectPotion()
+ addUnequippedItem(BasicEnemy): BasicItem
+ addInventoryItem(BasicItem)
+ getAllInventoryItems(): List<BasicItem>
+removeUnequippedInventoryItemByCoordinates(int, int)
+ removeUnequippedInventoryItem(Entity)
+ getUnequippedInventoryItemEntityByCoordinates(int, int): Entity
+ removeItemByPositionInUnequippedInventoryItems(int)
-getFirstAvailableSlotForItem(): Pair<Integer, Integer>
+ getRandomPosition(): Pair<Integer, Integer>
+ runBattles(LoopManiaWorldController): List<BasicEnemy>
+ getAllies(): List<Ally>
+getExperience():int
StaticEntity
- x : IntegerProperty
- y : IntegerProperty
+ StaticEntity (SimpleIntegerProperty, SimpleIntegerProperty, ImageView)
+ x() : IntegerProperty
+ y() : IntegerProperty
+ getX() : int
+ getY() : int
Entity
- shoudlExist : BooleanProperty
- image : ImageView
+ Entity(ImageView)
+ destroy() : void
+ shoudlExist() : BoolearnProperty
+ x() : IntegerProperty
+ y() : IntegerProperty
+ getX() : int
+ getY() : int
+ getImageView() : ImageView
MovingEntity
- position : PathPosition
- health : Health
+ MovingEntity(PathPosition, Health, ImageView)
+ moveDownPath() : void
+ moveUpPath() : void
+ x() : SimpleIntegerProperty
+ y() : SimpleIntegerProperty
+ getX() : int
+ getY() : int
+ getCurrentHealth(): int
+ decreaseHealth(int) : void
+ increaseHealth(int) : void
BasicItem
- costOfItem : int
+ BasicItem (SimpleIntegerProperty, SimpleIntegerProperty, int, ImageView)
+getCost() : int
Helmet
+ COST : int
+ Helmet (SimpleIntegerProperty, SimpleIntegerProperty)
+ getHelmet() : Helmet
+ reduceSlugDamage(Slug, Character) : void
+ reduceZombieDamage(Zombie, Character) : void
+reduceVampireDamage(Vampire, Character) : void
+reduceDoggieDamage(Doggie, Character): void
+reduceElanMuskeDamage (ElanMuske, Character)
Shield
+ COST : int
+ Shield (SimpleIntegerProperty, SimpleIntegerProperty)
+ getShield() : Shield
+ reduceSlugDamage(Slug, Character) : void
+ reduceZombieDamage(Zombie, Character) : void
+reduceVampireDamage(Vampire, Character) : void
+reduceDoggieDamage(Doggie, Character): void
+reduceElanMuskeDamage (ElanMuske, Character)
Armour
+ COST : int
+ Armour (SimpleIntegerProperty, SimpleIntegerProperty)
+ getArmour() : Armour
+ reduceSlugDamage(Slug, Character) : void
+ reduceZombieDamage(Zombie, Character) : void
+reduceVampireDamage(Vampire, Character) : void
+reduceDoggieDamage(Doggie, Character): void
+reduceElanMuskeDamage (ElanMuske, Character)
TreeStump
+ TreeStump (SimpleIntegerProperty, SimpleIntegerProperty)
+ getTreeStump() : TreeStump
+ reduceSlugDamage(Slug, Character) : void
+ reduceZombieDamage(Zombie, Character) : void
+reduceVampireDamage(Vampire, Character) : void
+reduceDoggieDamage(Doggie, Character): void
+reduceElanMuskeDamage (ElanMuske, Character)
<<Interface>>
DefendingStrategy
+ reduceSlugDamage(Slug, Character) : void
+ reduceZombieDamage(Zombie, Character) : void
+reduceVampireDamage(Vampire, Character) : void
+reduceDoggieDamage(Doggie, Character): void
+reduceElanMuskeDamage (ElanMuske, Character)
HealthPotion
+ COST : int
+ HealthPotion (SimpleIntegerProperty, SimpleIntegerProperty)
+ refillHealth(Character) : void
DoggieCoin
+ COST : int
+ DoggieCoin (SimpleIntegerProperty, SimpleIntegerProperty)
+ setCoinPrice(boolean):int
TheOneRing
+ TheOneRing (SimpleIntegerProperty, SimpleIntegerProperty)
+ reviveCharacter(Character) : void
Gold
+ goldCount : int
+ Gold (SimpleIntegerProperty, SimpleIntegerProperty)
+ getGoldCount() : int
+ increaseGold(int) : void
+ decreaseGold(int) : void
Ally
+ Ally(PathPosition, int)
Health
- currHealth: int
- maxHealth: int
+ Health(int)
+ getCurrHealth () : int
+ getMaxHealth () : int
+ decreaseHealth (int) : void
+ increaseHealth (int) : void
Observer Pattern
Observer Pattern
RegularState
+CharacterState(Character)
+dealDamage(BasicEnemy, int)
+ move()
StunState
+CharacterState(Character)
+dealDamage(BasicEnemy, int)
+ move()
JailState
+CharacterState(Character)
+dealDamage(BasicEnemy, int)
+ move()
CharacterState
-character. Character
+CharacterState(Character)
+dealDamage(BasicEnemy, int)
+ move()
State Pattern
Character
+ START_HEALTH: int
- basedamage : int
- gold : Gold
- experience : Experience
- equippedWeapon : AttackingStrategy
- equippedDefence : DefendingStrategy
- cardEntities : List<Card>
- allies : List<Ally>
- unequippedInventoryItems : List<BasicItem>
-unequipedRareItems: List<StaticEntity>
-state: CharacterState
+ Character(PathPosition)
+ getBaseDamage() : int
+ setBaseDamage(int) : void
+ getGoldCount() : int
+ increaseGold (Gold) : int
+ increaseGold (int) : int
+ decreaseGold (int) : int
+ decreaseGold (Gold) : int
+ getCharacterGold() : Gold
+ addExperience (int) : void
+ getExperience() : int
+ changeEquippedWeapon (AttackingStrategy) : void
+ getEquippedWeapon() : AttackingStrategy
+ changeEquippedDefence (DefendingStrategy) : void
+ getEquippedDefence() : DefendingStrategy
+ addCard (Card) : void
+ getCardByIndex (int) : Card
+ getAllCards() : List<Card>
+ addInventoryItem (Basicitem) : void
+ getAllInventoryItems() : ListBasicItem>
+addRareItem(StaticEntity):void
+getRareInventoryItems(): List<StaticEntity>
+ addAlly (Ally) : void
+ getAllies() : List<Ally>
+ sellItemByIndex(int) : BasicItem
+changeState(CharacterState):void
+getState():CharacterState
Battle
- enemiesToFight : List<BasicEnemy>
- c : Character
- controller : BattleEnemyController
- enemies : List<BasicEnemy>
- mainEnemy : BasicEnemy
- battleBuildings : List<BattleBuilding>
- loopCount : int
+ Battle(Character, BattleEnemyController, List<BasicEnemy>, BasicEnemy, List<BattleBuilding>, int)
+ dealDamageOnce() : void
+ getCharacter() : Character
+ getEnemiesToFight() : List<BasicEnemy>
+ killEnemy(BasicEnemy) : void
BattleEnemyController
-gameSwitcher: MenuSwitcher
-endSceneSwitcher: MenuSwitcher
-battle: Battle
-timeline: Time
+ BattleEnemyController()
+ setGameSwitcher(MenuSwitcher)
+ setEndSwitcher(MenuSwitcher)
+ switchToGame()
+ switchToEnd()
+ setBattle(Battle)
+ startTimer()
+ pauseBattle()
+ endGame()
PathPosition
- currentPositionInPath : int
- orderedPath : List<Pair<Integer, Integer>>
- x : SimpleIntegerProperty
- y : SimpleIntegerProperty
+ PathPosition(int, List<Pair<Integer, Integer>>)
+ moveDownPath() : void
+ moveUpPath() : void
+resetCoordinatesBaseOnPositionInPath() : void
+ getX() : SimpleIntegerProperty
+ getY() : SimpleIntegerProperty
+ getPreviousPathPosition(int) : PathPosition
LoopManiaWorldController
-squares: GridPane
-cards: GridPane
-anchorPaneRoot: AnchorPaneRoot
-equippedItems: GridPane
-unequippedInventory: GridPane
-healthNumber: Label
-goldNumber: Label
-expNumber: Label
-entityImagae: List<ImageView>
-draggedEntity:DragIcon
-isPaused: boolean
-world: LoopManiaWorld
-timeline: Timeline
-currentlyDraggedType: DRAGGABLE_TYPE
-mainMenuSwitcher: MenuSwitcher
-battleSwitcher:MenuSwitcher
-shopSellSwitcher
-battleEnemyController
-shopBuyController
-shopSellController
-healthNumber:Label
-goldNumber:Label
-expNumber:Label
-doggieGoal:CheckBox
-elanGoal:CheckBox
-goldGoal:CheckBox
-expGoal:CheckBox
-standardMode:Button
-survivalMode:Button
-berserkerMode:Button
-confusingMode:Button
*LoopManiaWorldController (LoopManiaWorld, List<ImageView>, BattleEnemyController, ShopSellController)
+ initialize()
+ startTimer()
+ pause()
+ terminate()
+ addEntity(entity, view)
+ loadCard(BasicEnemey)
+ loadItem(BasicEnemy)
+ reactToEnemyDefeat (BasicEnemy)
- onLoad(Card)
- onLoad(Basicitem)
- onLoad(Gold)
- onLoad(HealthPotion)
- onLoad(MovingEntity)
- onLoad(Building)
-buildNonEntityDragHandlers(DRAGGABLE_TYPE, GridPane, GridPane)
- convertCardToBuildingByCoordinates(int, int)
- removeItemByCoordinates(int, int)
-addFragEventHandlers(ImageView, DRAGGABLE_TYPE, Gridpane, Gridpane)
-removeDraggableDragEventHandlers(DRAGGABLE_TYPE, GridPane)
+ handleKeyPress(KeyEvent)
+ setMainMenySwitcher(MenuSwitcher)
+ setBattleSwitcher(MeanuSwitcher)
+ setShopSellSwotcher
+ switchToMainMenu()
+ switchToBattle
+ switchToShopSell
+ trackPosition(Entity, Node)
+ printThreadingNotes(String)
+ setGameRootParent()
+ getBattleController(): BattleEnemyController
+ getShopSellControll(): ShopSellController
+ getsUnequippedInventory(): GridPane
+ getRareItemsInventory():GridPane
+ getWorld():LoopManiaWorld
+ setStandardMode():void
+ setSurvivalMode():void
+ setBerserkerMode():void
+ setConfusingMode():void
EndScreenController
Experience
- experience : int
+ Experience()
+ addExperience(int) : void
+ getExperience() : int
ShopSellController
-inventory:GridPane
-buyItem:Button
-continueGame:Bbutton
-loopManiaWorld:LoopManiaWorld
-loopManiaWorldController: LoopManiaWorldController
-timeline:Timeline
+initialize():void
+setGameSwitcher(MenuSwitcher) :void
+startTimer(LoopManiaWorldController) :void
-switchToGame():void
-switchToShopBuy():void
+getItemByInventoryPosition(int, int): Basicitem
+setItem(Button):void
ShopBuyController
-itsitems:GridPane
-sellItem:Button
-gameSwitcher:MenuSwitcher
-shopSellSwitcher:MenuSwitcher
-timeLine:TimeLine
-initialize():void
-setGameSwitcher(MenuSwitcher menuSwitcher):void
-setShopSellSwitcher(ShopSellSwitcher shopSellSwitcher):void
-startTimer(LoopManiaWorldController loopManiaWorldController):void
-switchToGame():void
-switchToShopSell():void
MainMenuController
-gameSwitcher:MenuSwitcher
+setGameSwitcher(MenuSwitcher):void
-switchToGame():void
BasicEnemy
-attackRadius
-supportRadius
-damage
+BasicEnemy(PathPosition, Health, int, int, int, ImageView)
+getAttackRadius(): int
+getSupportRadius(): int
+getDamage(): int
+onDeath(SimpleIntegerProperty, SimpleIntegerProperty):StaticEntity
+move():void
+getGold():int
+getExperience():int
+dealDamage(DefendingStrategy, Character, List<BasicEnemy>):void
Slug
+Slug(PathPosition)
+onDeath(SimpleIntegerProperty, SimpleIntegerProperty) : StaticEntity
+onDeath(SimpleIntegerProperty, SimpleIntegerProperty):StaticEntity
+move():void
+getGold():int
+getExperience():int
+dealDamage(DefendingStrategy, Character, List<BasicEnemy>):void
Zombie
+ Zombie(PathPosition)
+onDeath(SimpleIntegerProperty, SimpleIntegerProperty):StaticEntity
+move():void
+getGold():int
+getExperience():int
+dealDamage(DefendingStrategy, Character, List<BasicEnemy>):void
Vampire
+ Vampire(PathPosition)
+onDeath(SimpleIntegerProperty, SimpleIntegerProperty):StaticEntity
+move():void
+getGold():int
+getExperience():int
+dealDamage(DefendingStrategy, Character, List<BasicEnemy>):void
ElanMuske
+ElanMuske(PathPosition)
+onDeath(SimpleIntegerProperty, SimpleIntegerProperty):StaticEntity
+move():void
+getGold():int
+getExperience():int
+dealDamage(DefendingStrategy, Character, List<BasicEnemy>):void
Doggie
+Doggie(PathPosition)
+onDeath(SimpleIntegerProperty, SimpleIntegerProperty):StaticEntity
+move():void
+getGold():int
+getExperience():int
+dealDamage(DefendingStrategy, Character, List<BasicEnemy>):void
Ghost
+level:int
+Ghost(PathPosition)
+onDeath(SimpleIntegerProperty, SimpleIntegerProperty):StaticEntity
+move():void
+getGold():int
+getExperience():int
+dealDamage(DefendingStrategy, Character, List<BasicEnemy>):void