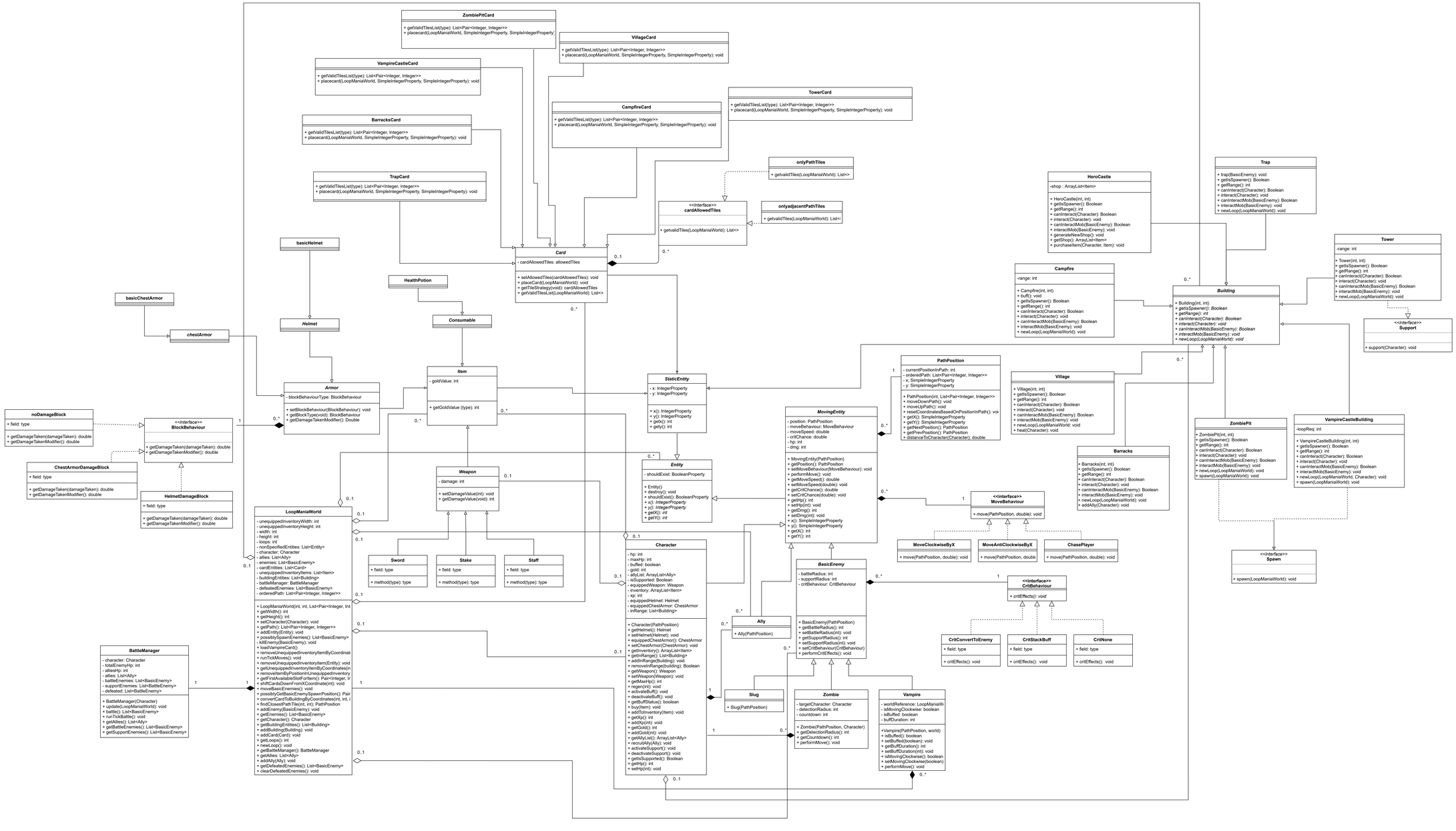
ZombiePitCard
+ getValidTilesList(type): List<Pair<Integer, Integer>>
+ placecard(LoopManiaWorld, SimpleIntegerProperty, SimpleIntegerProperty)
VillageCard
+ getValidTilesList(type): List<Pair<Integer, Integer>>
+ placecard(LoopManiaWorld, SimpleIntegerProperty, SimpleIntegerProperty): void
VampireCastleCard
+ getValidTilesList(type): List<Pair<Integer, Integer>>
+ placecard(LoopManiaWorld, SimpleIntegerProperty, SimpleIntegerProperty): void
TowerCard
+ getValidTilesList(type): List<Pair<Integer, Integer>>
+ placecard(LoopManiaWorld, SimpleIntegerProperty, SimpleIntegerProperty): void
CampfireCard
+ getValidTilesList(type): List<Pair<Integer, Integer>>
+ placecard(LoopManiaWorld, SimpleIntegerProperty, SimpleIntegerProperty): void
BarracksCard
+ getValidTilesList(type): List<Pair<Integer, Integer>>
+ placecard(LoopManiaWorld, SimpleIntegerProperty, SimpleIntegerProperty): void
TrapCard
+ getValidTilesList(type): List<Pair<Integer, Integer>>
+ placecard(LoopManiaWorld, SimpleIntegerProperty, SimpleIntegerProperty): void
onlyPathTiles
+ getvalidTiles(LoopManiaWorld): List<>
<<Interface>>
cardAllowedTiles
+ getvalidTiles(LoopManiaWorld): List<>
onlyadjacentPathTiles
+ getvalidTiles(LoopManiaWorld): List<
HeroCastle
-shop : ArrayList<Item>
+ HeroCastle(int, int)
+ getIsSpawner(): Boolean
+ getRange(): int
+ canInteract(Character): Boolean
+ interact(Character): void
+ canInteractMob(BasicEnemy): Boolean
+ interactMob(BasicEnemy): void
+ generateNewShop(): void
+ getShop(): ArrayList<Item>
+ purchaseItem(Character, Item): void
Trap
+ trap(BasicEnemy): void
+ getIsSpawner(): Boolean
+ getRange(): int
+ canInteract(Character): Boolean
+ interact(Character): void
+ canInteractMob(BasicEnemy): Boolean
+ interactMob(BasicEnemy): void
+ newLoop(LoopManiaWorld): void
Card
- cardAllowedTiles: allowedTiles
+ setAllowedTiles(cardAllowedTiles): void
+ placeCard(LoopManiaWorld): void
+ getTileStrategy(void): cardAllowedTiles
+ getValidTilesList(LoopManiaWorld): List<>
0..1
0..*
basicHelmet
HealthPotion
basicChestArmor
Helmet
Consumable
chestArmor
Campfire
-range: int
+ Campfire(int, int)
+ buff(): void
+ getIsSpawner(): Boolean
+ getRange(): int
+ canInteract(Character): Boolean
+ interact(Character): void
+ canInteractMob(BasicEnemy): Boolean
+ interactMob(BasicEnemy): void
+ newLoop(LoopManiaWorld): void
Tower
-range: int
+ Tower(int, int)
+ getIsSpawner(): Boolean
+ getRange(): int
+ canInteract(Character): Boolean
+ interact(Character): void
+ canInteractMob(BasicEnemy): Boolean
+ interactMob(BasicEnemy): void
+ newLoop(LoopManiaWorld): void
Building
+ Building(int, int)
+ getIsSpawner(): Boolean
+ getRange(): int
+ canInteract(Character): Boolean
+ interact(Character): void
+ canInteractMob(BasicEnemy): Boolean
+ interactMob(BasicEnemy): void
+ newLoop(LoopManiaWorld): void
<<Interface>>
Support
+ support(Character): void
Item
- goldValue: int
+ getGoldValue (type): int
Armor
- blockBehaviourType: BlockBehaviour
+ setBlockBehaviour(BlockBehaviour): void
+ getBlockType(void): BlockBehaviour
+ getDamageTakenModifier(): Double
StaticEntity
- x: IntegerProperty
- y: IntegerProperty
+ x(): IntegerProperty
+ y(): IntegerProperty
+ getx(): int
+ getyy(): int
PathPosition
- currentPositionInPath: int
- orderedPath: List<Pair<Integer, Integer>>
- x: SimpleIntegerProperty
- y: SimpleIntegerProperty
+ PathPosition(int, List<Pair<Integer, Integer>>
+ moveDownPath(): void
+ moveUpPath(): void
+ resetCoordinatesBasedOnPositionInPath(): void
+ getX(): SimpleIntegerProperty
+ getY(): SimpleIntegerProperty
+ getNextPosition(): PathPosition
+ getPrevPosition(): PathPosition
+ distanceToCharacter(Character): double
1
0..*
Village
+ Village(int, int)
+ getIsSpawner(): Boolean
+ getRange(): int
+ canInteract(Character): Boolean
+ interact(Character): void
+ canInteractMob(BasicEnemy): Boolean
+ interactMob(BasicEnemy): void
+ newLoop(LoopManiaWorld: void
+ heal(Character): void
noDamageBlock
+ field: type
+ getDamageTaken(damageTaken): double
+ getDamageTakenModifier(): double
<<Interface>>
BlockBehaviour
+ getDamageTaken(damageTaken): double
+ getDamageTakenModifier(): double
ChestArmorDamageBlock
+ field: type
+ getDamageTaken(damageTaken): double
+ getDamageTakenModifier(): double
HelmetDamageBlock
+ field: type
+ getDamageTaken(damageTaken): double
+ getDamageTakenModifier(): double
Weapon
- damage: int
+ setDamageValue(int): void
+ getDamageValue(void): int
Entity
- shouldExist: BooleanProperty
+ Entity()
+ destroy(): void
+ shouldExist(): BooleanProperty
+ x(): IntegerProperty
+ y(): IntegerProperty
+ getX(): int
+ getY(): int
MovingEntity
- position: PathPosition
- moveBehaviour: MoveBehaviour
- moveSpeed: double
- critChance: double
- hp: int
- dmg: int
+ MovingEntity(PathPosition)
+ getPosition(): PathPosition
+ setMoveBehaviour(MoveBehaviour): void
+ performMove(): void
+ getMoveSpeed(): double
+ setMoveSpeed(double): void
+ getCritChance(): double
+ setCritChance(double): void
+ getHp(): int
+ setHp(int): void
+ getDmg(): int
+ setDmg(int): void
+ x(): SimpleIntegerProperty
+ y(): SimpleIntegerProperty
+ getX(): int
+ getY(): int
0..*
1
<<interface>>
MoveBehaviour
+ move(PathPosition, double): void
Barracks
+ Barracks(int, int)
+ getIsSpawner(): Boolean
+ getRange(): int
+ canInteract(Character): Boolean
+ interact(Character): void
+ canInteractMob(BasicEnemy): Boolean
+ interactMob(BasicEnemy): void
+ newLoop(LoopManiaWorld): void
+ addAlly(Character): void
ZombiePit
+ ZombiePit(int, int)
+ getIsSpawner(): Boolean
+ getRange(): int
+ canInteract(Character): Boolean
+ interact(Character): void
+ canInteractMob(BasicEnemy): Boolean
+ interactMob(BasicEnemy): void
+ newLoop(LoopManiaWorld): void
+ spawn(LoopManiaWorld): void
VampireCastleBuilding
-loopReq: int
+ VampireCastleBuilding(int, int)
+ getIsSpawner(): Boolean
+ getRange(): int
+ canInteract(Character): Boolean
+ interact(Character): void
+ canInteractMob(BasicEnemy): Boolean
+ interactMob(BasicEnemy): void
+ newLoop(LoopManiaWorld, Character): void
+ spawn(LoopManiaWorld): void
<<Interface>>
Spawn
+ spawn(LoopManiaWorld): void
LoopManiaWorld
- unequippedInventoryWidth: int
- unequippedInventoryHeight: int
- width: int
- height: int
- loops: int
- nonSpecifiedEntities: List<Entity>
- character: Character
- allies: List<Ally>
- enemies: List<BasicEnemy>
- cardEntities: List<Card>
- unequippedInventoryItems: List<Item>
- buildingEntities: List<Building>
- battleManager: BattleManager
- defeatedEnemies: List<BasicEnemy>
- orderedPath: List<Pair<Integer, Integer>>
+ LoopManiaWorld(int, int, List<Pair<Integer, Int
+ getWidth(): int
+ getHeight(): int
+ setCharacter(Character): void
+ getPath() : List<Pair<Integer, Integer>>
+ addEntity(Entity): void
+ possiblySpawnEnemies(): List<BasicEnemy>
- killEnemy(BasicEnemy): void
+ loadVampireCard()
+ removeUnequippedInventoryItemByCoordinat
+ runTickMoves(): void
+ removeUnequippedInventoryItem(Entity): void
+ getUnequippedInventoryItemByCoordinates(in
+ removeItemByPositionInUnequippedInventory
+ getFirstAvailableSlotForItem(): Pair<Integer, In
+ shiftCardsDownFromXCoordinate(int): void
+ moveBasicEnemies(): void
+ possiblyGetBasicEnemySpawnPosition(): Pair
+ convertCardToBuildingByCoordinates(int, int, i
+ findClosestPathTile(int, int): PathPosition
+ addEnemy(BasicEnemy): void
+ getEnemies() : List<BasicEnemy>
+ getCharacter(): Character
+ getBuildingEntities() : List<Building>
+ addBuilding(Building): void
+ addCard(Card): void
+ getLoops(): int
+ newLoop(): void
+ getBattleManager(): BattleManager
+ getAllies: List<Ally>
+ addAlly(Ally): void
+ getDefeatedEnemies(): ListBasicEnemy>
+ clearDefeatedEnemies(): void
Sword
+ field: type
+ method(type): type
Stake
+ field: type
+ method(type): type
Staff
+ field: type
+ method(type): type
Character
- hp: int
- maxHp: int
- buffed: boolean
- gold: int
- allyList: ArrayList<Ally>
- isSupported: Boolean
- equippedWeapon: Weapon
- inventory: ArrayList<Item>
- xp: int
- equippedHelmet: Helmet
- equippedChestArmor: ChestArmor
- inRange: List<Building>
+ Character(PathPosition)
+ getHelmet(): Helmet
+ setHelmet(Helmet): void
+ equippedChestArmor(): ChestArmor
+ setChestArmor(ChestArmor): void
+ getInventory() : ArrayList<Item>
+ getInRange() : List<Building>
+ addInRange(Building): void
+ removeInRange(building): Boolean
+ getWeapon(): Weapon
+ setWeapon(Weapon): void
+ getMaxHp(): int
+ regen(int): void
+ activateBuff(): void
+ deactivateBuff(): void
+ getBuffStatus(): boolean
+ buy(Item): void
+ addToInventory(Item): void
+ getXp(): int
+ addXp(int): void
+ getGold(): int
+ addGold(int): void
+ getAllyList(): ArrayList<Ally>
+ recruitAlly(Ally): void
+ activateSupport(): void
+ deactivateSupport(): void
+ getIsSupported(): Boolean
+ getHp(): int
+ setHp(int): void
0..1
BasicEnemy
- battleRadius: int
- supportRadius: int
- critBehaviour: CritBehaviour
+ BasicEnemy(PathPosition)
+ getBattleRadius(): int
+ setBattleRadius(int): void
+ getSupportRadius(): int
+ setSupportRadius(int): void
+ setCritBehaviour(CritBehaviour)
+ performCritEffects(): void
<<interface>>
CritBehaviour
+ critEffects(): void
CritConvertToEnemy
+ field: type
+ critEffects(): void
CritStackBuff
+ field: type
+ critEffects(): void
CritNone
+ field: type
+ critEffects(): void
Ally
+ Ally(PathPosition)
Slug
+ Slug(PathPosition)
Zombie
- targetCharacter: Character
- detectionRadius: int
- countdown: int
+ Zombie(PathPosition, Character)
+ getDetectionRadius(): int
+ getCountdown(): int
+ performMove() : void
Vampire
- worldReference: LoopManiaW
- isMovingClockwise: boolean
- isBuffed: boolean
- buffDuration: int
+Vampire(PathPosition, world)
+ isBuffed(): boolean
+ setBuffed(boolean): void
+ getBuffDuration(): int
+ setBuffDuration(int): void
+ isMovingClockwise(): boolean
+ setMovingClockwise(boolean)
+ performMove(): void
MoveClockwiseByX
+ move(PathPosition, double): void
MoveAntiClockwiseByX
+ move(PathPosition, double
ChasePlayer
+ move(PathPosition, double): void
BattleManager
- character: Character
- totalEnemyHp: int
- alliesHp: int
- allies: List<Ally>
- battleEnemies: List<BasicEnemy>
- supportEnemies: List<BattleEnemy>
- defeated: List<BattleEnemy>
+ BattleManager(Character)
+ update(LoopManiaWorld): void
+ battle() : List<BasicEnemy>
+ runTickBattle(): void
+ getAllies() : List<Ally>
+ getBattleEnemies() : List<BasicEnemy>
+ getSupportEnemies() : List<BasicEnemy>
1
0..1