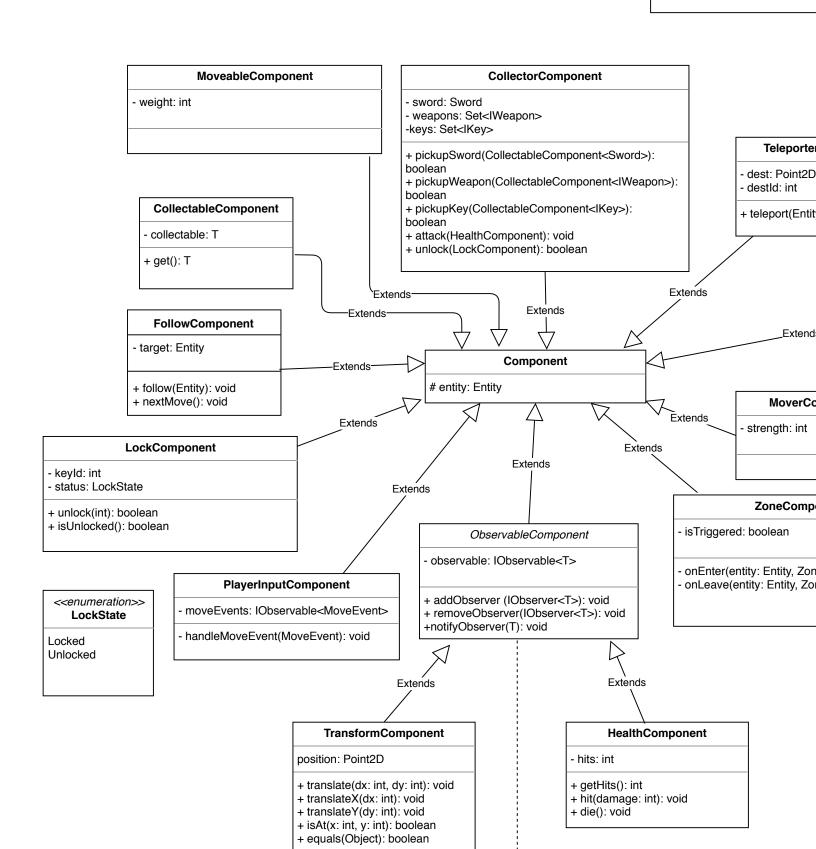
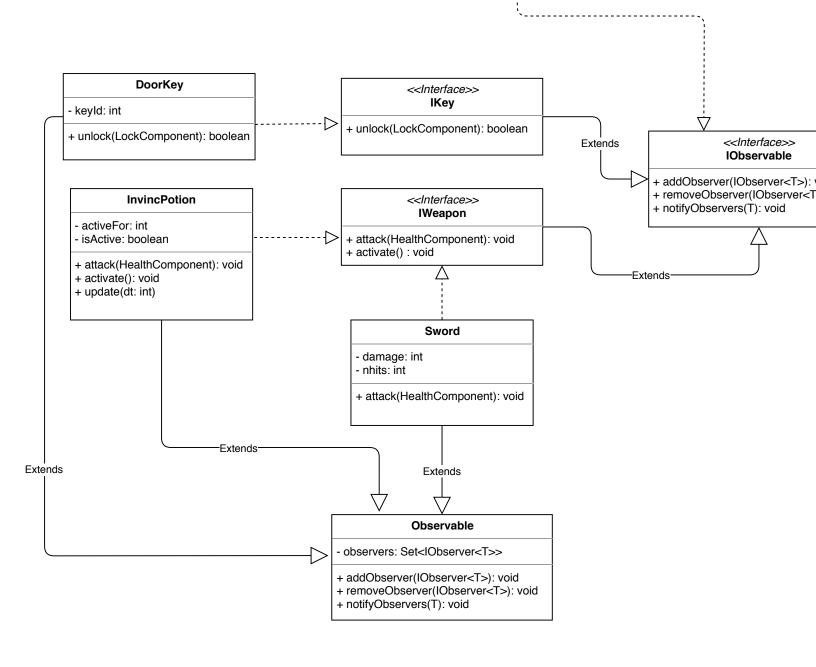
## - entityManager: EntityManager - goal: GoalComponent + start() + pause() + resume() + restart()



		MoveEvent	
		+ direction: Direction	< <enumeration></enumeration>
		+ up(): MoveEvent + right(): MoveEvent + down(): MoveEvent+ left():	UP RIGHT DOWN LEFT
		< <interface: +="" iobserver="" td="" update(t):="" void<=""><td></td></interface:>	
component			
): void			
	Entity	Delta	
	- components: Set <component> - name: String</component>	+ from: T + to: T	
mponent	+ getComponent(type: Class): Component + addComponent(component: Component): Component + removeComponent(type: Class): void +addCollisionStrategy(entity: Entity, strategy: CollisionStrategy)	<pre>&lt;<interface>&gt; CollisionStrategy + exec(Delta<point2d>): void</point2d></interface></pre>	
		Point2D	
nent	EntityFactory	- x: int	
	+ BOULDER_WEIGHT: int + SWORD_DAMANGE: int	- y: int	
eHandler): void eHandler): void	+ SWORD_NHITS : int +TREASURE_VALUE: int + INVINC_DURATION_MS: int + UID: int	+ equals(Object): boolear	
	- withUID(id: String): String + createPlayer(int, int, IObservable): Entity + createWall(int, int): Entity + createExit(int, int): Entity + createTreasure(int, int): Entity + createDoor(int, int, int): Entity + createKey(int, int, int): Entity + createBoulder(int, int): Entity + createFloorSwitch(int, int): Entity + createPortal(int, int, int): Entity + createEnemy(int, int): Entity + createSword(int, int): Entity + createInvincibilityPotion(int, int): Entity	Treasure - value: int	



void >): void