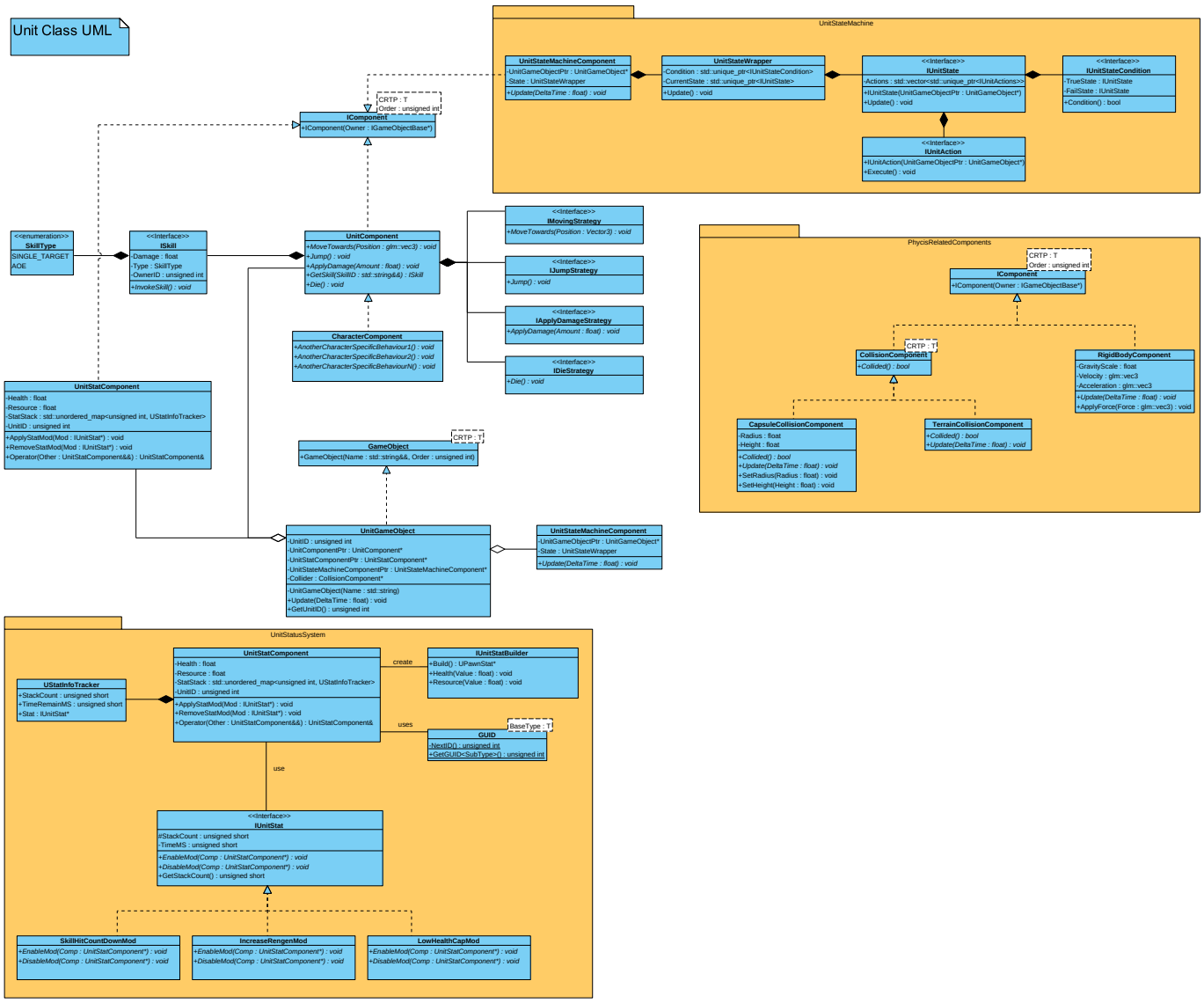
Unit Class UML
UnitStateMachine
UnitStateMachineComponent
-UnitGameObjectPtr : UnitGameObject*
-State : UnitStateWrapper
+Update(DeltaTime : float) : void
UnitStateWrapper
-Condition : std::unique_ptr<IUnitStateCondition>
-CurrentState : std::unique_ptr<IUnitState>
+Update() : void
<<Interface>>
IUnitState
-Actions : std::vector<std::unique_ptr<IUnitActions>>
+IUnitState(UnitGameObjectPtr : UnitGameObject*)
+Update() : void
<<Interface>>
IUnitStateCondition
-TrueState : IUnitState
-FailState : IUnitState
+Condition() : bool
<<Interface>>
IUnitAction
+IUnitAction(UnitGameObjectPtr : UnitGameObject*)
+Execute() : void
CRTP : T
Order : unsigned int
IComponent
+IComponent(Owner : IGameObjectBase*)
<<enumeration>>
SkillType
SINGLE_TARGET
AOE
<<Interface>>
ISkill
-Damage : float
-Type : SkillType
-OwnerID : unsigned int
+InvokeSkill() : void
UnitComponent
+MoveTowards(Position : glm::vec3) : void
+Jump() : void
+ApplyDamage(Amount : float) : void
+GetSkill(SkillID : std::string&&) : ISkill
+Die() : void
<<Interface>>
IMovingStrategy
+MoveTowards(Position : Vector3) : void
<<Interface>>
IJumpStrategy
+Jump() : void
<<Interface>>
IApplyDamageStrategy
+ApplyDamage(Amount : float) : void
<<Interface>>
IDieStrategy
+Die() : void
CharacterComponent
+AnotherCharacterSpecificBehaviour1() : void
+AnotherCharacterSpecificBehaviour2() : void
+AnotherCharacterSpecificBehaviourN() : void
UnitStatComponent
-Health : float
-Resource : float
-StatStack : std::unordered_map<unsigned int, UStatInfoTracker>
-UnitID : unsigned int
+ApplyStatMod(Mod : IUnitStat*) : void
+RemoveStatMod(Mod : IUnitStat*) : void
+Operator(Other : UnitStatComponent&&) : UnitStatComponent&
GameObject
+GameObject(Name : std::string&&, Order : unsigned int)
CRTP : T
UnitGameObject
-UnitID : unsigned int
-UnitComponentPtr : UnitComponent*
-UnitStatComponentPtr : UnitStatComponent*
-UnitStateMachineComponentPtr : UnitStateMachineComponent*
-Collider : CollisionComponent*
-UnitGameObject(Name : std::string)
+Update(DeltaTime : float) : void
+GetUnitID() : unsigned int
UnitStateMachineComponent
-UnitGameObjectPtr : UnitGameObject*
-State : UnitStateWrapper
+Update(DeltaTime : float) : void
PhycisRelatedComponents
CRTP : T
Order : unsigned int
IComponent
+IComponent(Owner : IGameObjectBase*)
CRTP : T
CollisionComponent
+Collided() : bool
RigidBodyComponent
-GravityScale : float
-Velocity : glm::vec3
-Acceleration : glm::vec3
+Update(DeltaTime : float) : void
+ApplyForce(Force : glm::vec3) : void
CapsuleCollisionComponent
-Radius : float
-Height : float
+Collided() : bool
+Update(DeltaTime : float) : void
+SetRadius(Radius : float) : void
+SetHeight(Height : float) : void
TerrainCollisionComponent
+Collided() : bool
+Update(DeltaTime : float) : void
UnitStatusSystem
UStatInfoTracker
+StackCount : unsigned short
+TimeRemainMS : unsigned short
+Stat : IUnitStat*
UnitStatComponent
-Health : float
-Resource : float
-StatStack : std::unordered_map<unsigned int, UStatInfoTracker>
-UnitID : unsigned int
+ApplyStatMod(Mod : IUnitStat*) : void
+RemoveStatMod(Mod : IUnitStat*) : void
+Operator(Other : UnitStatComponent&&) : UnitStatComponent&
create
IUnitStatBuilder
+Build() : UPawnStat*
+Health(Value : float) : void
+Resource(Value : float) : void
uses
BaseType : T
GUID
-NextID() : unsigned int
+GetGUID<SubType>() : unsigned int
use
<<Interface>>
IUnitStat
#StackCount : unsigned short
-TimeMS : unsigned short
+EnableMod(Comp : UnitStatComponent*) : void
+DisableMod(Comp : UnitStatComponent*) : void
+GetStackCount() : unsigned short
SkillHitCountDownMod
+EnableMod(Comp : UnitStatComponent*) : void
+DisableMod(Comp : UnitStatComponent*) : void
IncreaseRengenMod
+EnableMod(Comp : UnitStatComponent*) : void
+DisableMod(Comp : UnitStatComponent*) : void
LowHealthCapMod
+EnableMod(Comp : UnitStatComponent*) : void
+DisableMod(Comp : UnitStatComponent*) : void