AMainPlayer

This is a UML class diagram showing the class **AMainPlayer** inheriting from **ACharacter** and **IGenericTeamAgentInterface**, with a relationship to **IPlayerNotifier**.

AMainPlayer

Attributes:
- + TArray< AAbstractItem * > PickUpItems
- + TSubclassOf< UCommandWidget > CommandWidgetClass
- + TSubclassOf< UPlayerHUD > PlayerHUDWidgetClass
- + UPlayerHUD * PlayerHUDWidget
- + TSubclassOf< UServerWidget > ServerWidgetClass
- + UServerWidget * ServerWidget
- + FSetStats SetStats
- + FDifficultyModeSet SetDifficultyModeDelegate
- + FOnCharacterJoinSession OnCharacterJoinSession
- + FAnimNotify_Jump AnimNotify_Jump
- + TArray< IEnemyObserver * > EnemyObservers
- # UCameraComponent * CameraComponent
- # float InteractDistance
- # bool bContinuable
- # FVector PreviousActorLocation
- # FRotator PreviousActorRotation
- # FVector WeaponChooseLocation
- # FRotator WeaponChooseRotation
- # bool bShowCrosshair
- # TArray< AWeaponClass * > PickedWeapons
- # float TurnRate
- # float LookUpRate
- # float MouseSensivity
- # TSubclassOf< AWeaponClass > AwmClass
- # TSubclassOf< AWeaponClass > M16A4Class
- # TSubclassOf< AWeaponClass > Ak47Class
- # TSubclassOf< AWeaponClass > M870Class
- # TSubclassOf< AWeaponClass > Hk416Class
- # TArray< FString > Player WeaponSocketsName
- # AWeaponClass * EquippedWeapon
- # IWeaponInterface * EquippedDifficultyWeapon
- # FString SPlayerName
- # FString SCity
- # bool IsInGodMode
- # FBehaviorSet BehaviorSet
- # float Health
- # int HealthState
- # float Armor
- # int PlayerIndex
- # TSubclassOf< UUserWidget > CrosshairWidgetClass
- # TSubclassOf< UWeaponMenu Widget > WeaponMenuWidgetClass
- # UUserWidget * CrosshairWidget
- # UWeaponMenuWidget * WeaponMenuWidget
- # TSubclassOf< UMainMenuWidget > MainMenuWidgetClass
- # UMainMenuWidget * MainMenu Widget

Operations:
- + virtual void SetupPlayer InputComponent(class UInputComponent *PlayerInputComponent) override
- + virtual void Tick(float DeltaSeconds) override
- + void OnBeginOverlap (UPrimitiveComponent *OverlappedComponent, AActor *OtherActor, UPrimitive Component *OtherComp, int OtherBody Index, bool FromSweep, const FHitResult &SweepResult)
- + void MoveForward(float Scale)
- + void MoveRight(float Scale)
- + virtual void Jump() override
- + UCameraComponent * GetCameraComponent()
- + void SetHealth()
- + void SetArmor()
- + void SetHealthArmor()
- + void ActivateBuffs (const TScriptInterface < IHandler > &Handler)
- + void Server_Turn(float Rate)
- + void Multicast_Turn (float Rate)
- + void Server_LookUp (float Rate)
- + void Multicast_LookUp (float Rate)
- + void Interact()
- + void ContinueGameplay()
- + void ProcessHitRack()
- + void ProcessHitWeapon (AWeaponClass *WeaponActor)
- + void RotatePositive (AActor *Actor)
- + void RotateNegative (AActor *Actor)
- + void OnCommandWidgetClosed()
- + void ProcessHitCube (AActor *HitActor)
- + TArray< AWeaponClass * > GetAllPickedWeapons()
- + void Shoot()
- + void Server_Shoot(FVector StartTrace, FVector EndTrace)
- + void Multicast_Shoot (FVector StartTrace, FVector EndTrace)
- + void GetDamage(int Damage)
- + void AddHealth(float HealthPoints)
- + void AddArmor(float ArmorPoints)
- and 34 more...
- # AMainPlayer()
- # virtual void BeginPlay () override
- # void CreateServerWidget()
- # void CreateHUDWidget()
- # void CreateMainMenuWidget()
- # void SetDifficultyMode (const FString &Difficulty)

ACharacter

IGenericTeamAgentInterface

IPlayerNotifier

- + virtual void Attach (IEnemyObserver *EnemyInterface)=0
- + virtual void Detach (IEnemyObserver *EnemyInterface)=0
- + virtual void Notify()=0