

[illegible]

```

graph LR
    subgraph Propagate
        direction TB
        P_damage[+ damage float]
        P_hit[+ hit bool]
        P_choose1[+ chooseDamage float -> 0]
        P_choose2[+ chooseDamage float -> 0]
        P_choose2 --> P_targetFlow[+ targetFlow string -> "Tranny", "Tranny"]
    end

    subgraph ADEvent
        direction TB
        A_damage[+ damage float]
        A_choose1[+ chooseDamage float -> 0]
        A_choose2[+ chooseDamage float -> 0]
        A_choose2 --> A_targetFlow[+ targetFlow string -> "Tranny", "Tranny"]
        A_Start[+ Start -> actual void]
        A_End[+ End -> actual void]
        A_choose3[+ chooseDamage float -> 0]
        A_choose3 --> A_targetFlow
        A_Decrypt[+ Decrypt[Self] -> void]
    end

    P_damage --> A_damage
    P_hit --> A_choose1
    P_choose1 --> A_choose2
    P_choose2 --> A_choose3
    P_targetFlow --> A_targetFlow
    A_Start --> A_choose3
    A_End --> A_choose3
    A_Decrypt --> A_choose3
  
```

[illegible][illegible]

MonoBehavior

[illegible]

AnchorType.h	LockPosition
<ul style="list-style-type: none"> • headShared DecTransform • after: Vector2 	<ul style="list-style-type: none"> • initPos: Vector2
<ul style="list-style-type: none"> • Update() void 	<ul style="list-style-type: none"> • Start() void • LaunchShot() void

```

classDiagram
    class Builder {
        None
        OnCollisionEnterCallback callback() void
    }
    class CharacterSelectionMenu {
        + characterSelectionPanel: GameObject
        + characterIndex: GameObject
        + PFHeld: GameObject
        + characterName: Transform
    }
    class CharacterSpawner {
        + spawnRate: Transform
        Start() void
    }
    class GameManager {
        + instance: GameManager
        + spawnCharacter: GameObject
        + gameOverPanel: GameObject
        + gameOverText: Text
        + victory: Button
        + defeat: Button
    }
    class GunBehaviour {
        + bulletPrefab: GameObject
        + bulletPrefab: Transform
        + bulletCount: int
        + bulletSpeed: float = 20f
        + Update() void
        + Shoot() void
    }
    class GunManager {
        + guns: GameObject[]
        + currentGunIndex: int = 0
    }
    class PlayerController {
        + healthMaxSpeed: float = 2
        + maxCamera: Camera
        + Rigidbody: Rigidbody
        + movement: Vector2
        + wander: Vector2
        + playerClass: MonoBehaviour
        + playerClass: MonoBehaviour
    }
    Builder --> CharacterSelectionMenu
    Builder --> CharacterSpawner
    Builder --> GameManager
    Builder --> GunBehaviour
    Builder --> GunManager
    Builder --> PlayerController
    
```

```

classDiagram
    class CookieManager {
        +cookieStorage: Image
        +processAuthCookie: Image
        +probability: Sprite
        +cookieDuration: float
        +cookieRemaining: float
        +cookieActive: bool + false
    }
    class SetAbilityCookiePlayerClass {
        +SetAbilityCookiePlayerClass: string; void
        +StartCookieDuration: float; void
        +Update: void
    }
    CookieManager --> SetAbilityCookiePlayerClass
  
```

[illegible][illegible]