

## Sigilspire Early Prototype — Designer-Friendly Checklist

### 1. Visual Identity

- Player requires 8-direction idle and run sprites.
- Weapons need identifiable idle states.
- Sword needs 4-direction slash animations.
- Shield needs a block pose and break effect.
- Grapple needs a cast and hit visual.
- VFX should clearly communicate action impact.

### 2. Animation Controllers

- PlayerAnimator handles core movement identity.
- WeaponAnimator runs weapon actions (slash, block, grapple).
- Each sigil should have an AnimatorOverrideController.
- Overrides allow easy visual swaps per sigil theme.

### 3. Core Prefab Structure

#### Player

- VisualRoot (all visual identity lives here)
  - PlayerAnimator — body animation
  - WeaponAnimator — combat animation
  - BodyRenderer — player's core sprite
  - WeaponRenderers — all weapon sprites grouped
  - SwordRenderer
  - ShieldRenderer
  - GrappleRenderer
- Logic-only weapon objects:
  - Sword (SwordController)

■ ■ ■ Shield (ShieldController)

■ ■ ■ Grapple (GrapplingHookController)

#### 4. Sigil Visual System

- Each sigil defines its own visuals via a WeaponVisualSet.
- VisualSet includes override controller, sprites, VFX, and audio.
- Changing sigils changes the weapon's entire identity in one step.

#### 5. Gameplay Hooks

- Loadout selection passes sigils into the PlayerController.
- WeaponControllers load visuals and stats from sigils.
- Visuals, effects, and animations respond instantly to loadout swaps.

#### 6. Playtesting Goals

- Combat feels responsive and readable.
- Player direction and weapon direction align visually.
- Visual feedback is clear (hits, blocks, grapples).
- Weapon identity reflects chosen sigil theme.