**FLICKER – INITIAL GDD**

**Core Elements**

* Fast, fluid, involved movement
* Emphasis on speed & momentum with a sprinkling of resource management
* Light/Shadow-based mechanics/themes
* Interact with world through “light-knife”
* Light/Shadow environmental puzzles
* Player initially ‘powered’ by light
* Restrictions – eg: player can only warp to knife if it’s in light; lose access to some abilities when in shadow
* Knife has a number of energy charges
* A charge is used up each time the player warps or performs special actions
* Charges recharge at a moderate pace, and only when the player is in a lit area
* Recharge is slowed significantly while in mid-air
* Game world state preserved on ‘death’
* Preserve found upgrades etc by ‘saving’ – backing up current Script state – at set locations
* New ‘lives’ are your previously backed-up ‘Script’ being written into a blank body
* Default Script extension causes player to turn to crystal/glass on death – preserving current state
* Can reclaim any lost upgrades etc by reaching site of death
* Minimal/No in-game UI
* Knife charges indicated by glowing Script elements on in-game model
* Progression by finding Script extensions in world
* Gives new abilities, or extends existing ones (eg: short glide vs adding an extra knife charge)
* Start with a set of extensions installed
* Act like chip system from Nier – Limit based on available skin area for writing
* EVERYTHING can be installed/uninstalled, from knife charge indicators to fall damage prevention to the preservation of your body on death
* Game consists of light/traversal-based environmental puzzles and challenges; and combat designed to require quick movement and careful use of resources
* Both player and normal enemies can take very little damage
* Normal enemies die in one hit when stuck with knife and then warped to – restoring a knife charge in the process
* Enemy attacks will kill player very quickly, focus on quick evasion rather than tanking hits
* Enemies used to encourage player to move quickly
* Combat scenarios treated more as environmental obstacles – eg: static defenses

Story/Universe Premise

* ‘Script’ is an in-universe written divine language that defines the physical properties of anything it is written on – powered by light?
* The player is a homunculus created/invented by a certain ‘god’ by applying Script to a blank body
* Current player state is represented by Script that covers body
* Both light and dark god-creatures exist, and are visible and interact-able under certain conditions (world state – possibly certain abilities)
* Otherworldly
* (Some?) can be killed by the player

INITIAL STUFF

Light Gameplay

* Power comes from light?
  + Lose all abilities except basic movement when in shadow?
  + Can’t easily throw knife through shadows?
  + Knife could instantly return, or lose all momentum and player must retrieve it manually?
  + Gravity switching required to illuminate shadowed areas?
  + Or: Player cannot leave lit areas, but knife can
  + OR: Abilities only recharge in light!
  + Possible slow decay of energy in shadow?
* Knife acts as light when thrown
* Light-knife interacts semi-realistically with transparent/mirrored surfaces
  + Knife passes through glass surfaces while player cannot
  + ‘fibre-optic’ style cables can transport knife around curves etc.

**Basic Gameplay Details**

Controls

* First-person
* Standard WASD movement + Jump
* Sprint optional
* LMB throws ‘default’ knife
* RMB returns default knife if thrown, else throws ‘bouncy’ knife

Basic Movement & Momentum

* Player control must be responsive but take into account velocity & momentum.
* Air movement must follow momentum by default but still allow corrections/adjustment
* Sprinting downhill allows player to pick up momentum – uphill slows player (although less than downhill gains)
* By sprinting downhill and jumping/using knife movement uphill player can gain and keep speed and momentum – allows skilful players to move quickly over long distances.
* Similarities to 2D phone games involving using downhill slopes to gain momentum

Knife Basics & Interactions

* Left click will throw a ‘default’ knife
* Right click throws ‘alternate’ knife – currently the bouncing knife
* Knives travel very fast – allows for accuracy at a distance and just feels better. Nobody wants a sluggish knife.
* Default knife travels for set time before automatically returning – limits range of player interaction
* Default knife will stick into most surfaces on contact and remain indefinitely (may need range/time limit on this)
* LMB while a knife is stuck in a surface will warp the player to the knife’s position
* Bouncy knife travels a short distance before warping the player to its position
* If bouncy knife collides with a surface it will reflect and extend the timer before triggering the warp

**Environment**

* Mix of sharp/angular surfaces and smooth, rolling landscapes
* Angular surfaces allow simpler ricochet/bounce prediction
* Smooth/curved landscapes enable momentum conservation
* Environment designed to encourage & facilitate creative use of player movement mechanics
* Surfaces can be shiny/glossy or matte to distinguish knife behaviour
* Shiny surfaces always reflect knife
* Matte surfaces do not reflect knife (possible exception for bouncing knife)
* A lot of glass elements in environment
* Possible inclusion of mirrors