

UNKINDNESS OF RAVENS

Portfolio Case Study · Game & Narrative Systems Design

Role Solo Game & Narrative Designer

Genre Narrative Exploration · Action RPG · Platformer

Engine Unreal Engine 5 — Blueprints

Status In Development — Demo targeting IGDC 2026, Chennai

OVERVIEW

UKoR is set in a dying pre-apocalyptic world. A fraction of the population can escape aboard massive evacuation ships — Ravens. The player is a rookie crow pilot flying a steampunk mechanical familiar across a crumbling landscape of cities, villages, and forgotten vaults. When the Raven launches, the story ends. Whether you've earned your seat is the game.

DESIGN PROBLEM 1 — MAKING THE DEADLINE FEEL REAL

The game's central tension is time. A hard countdown to the Raven's departure defines the world. The design challenge: how do you make players feel the weight of that timer without it becoming punishing or frustrating?

A ticking clock that exists only as a UI counter is anxiety. A ticking clock that changes the world is stakes.

The solution was to make the deadline diegetic. Rather than a HUD timer, the Raven's departure is woven into world state: docks get busier, prices shift, NPCs begin making their own evacuation plans, certain quests expire. The player feels time passing through the world's behavior, not a number.

- World State Variables flag which NPCs have left, which routes have closed, which factions have started prioritising themselves over the player
- Quests that are time-sensitive surface urgency through NPC dialogue and environmental cues — not timers
- The player's reputation with factions determines how much runway they have: high trust buys you time, low trust accelerates abandonment

DESIGN PROBLEM 2 — DUAL LOCOMOTION: GROUND CHARACTER + DRONE FLIGHT

The player controls two bodies: a ground character and a mechanical familiar (the Feather). Each has a distinct control philosophy. The design problem was ensuring the switch between them felt intentional and meaningful, not disorienting.

The ground character moves within a third-person action framework — dodge, dash, interact, combat. The Feather operates on a flight model referencing real FPV drone control, with pitch, roll, yaw, and throttle as independent axes across four configurable stick modes.

Context	Ground (IMC_Player_Ground)	Drone Flight (IMC_DroneFlight)
Primary input	WASD / Left Stick — directional movement	Pitch/Roll/Yaw/Throttle — 4 independent axes
Core feel	Grounded, weight, collision response	Aerial, momentum, inertia
Switch trigger	G / Y / Δ — toggle flight mode	Same — returns to ground character
Priority	IMC_Player_Ground = 0	IMC_DroneFlight = 100 (overrides)

The Input Mapping Context (IMC) priority system resolves conflicts: when the drone is active, IMC_DroneFlight at priority 100 overrides IMC_Player_Ground at 0. The player always knows which body is active without a mode indicator — the feel of the inputs tells them.

DESIGN PROBLEM 3 — KOEL: CONSEQUENCE WITHOUT A MORALITY METER

UKoR contains a faction-reactivity system called Koel. The design mandate was clear from the start: the world should respond to player choices without a visible morality bar. Meters flatten nuance — the player optimises for the score, not the story.

"The island remembers" — Koel's core constraint. Every action happens in a space where people saw it, heard about it, or will feel its consequences.

Koel operates across four interconnected layers:

- **Faction Disposition** — Faction Disposition
 - a float from -100 to +100 per faction, modified by events not accumulation. Five behavioral tiers (Trusted → Enemy) that manifest as changed dialogue, prices, access, and NPC behavior — never as a visible number
- **World State Variables** — World State Variables
 - permanent flags that record what has changed in the world. These never decay. A burned warehouse stays burned
- **NPC Memory** — NPC Memory
 - per-NPC knowledge that decays unless reinforced. A guard who saw you steal forgets over time — unless his colleague reminds him
- **District Evolution** — District Evolution
 - physical spaces change in response to accumulated player actions. The feedback is spatial, not numerical

Faction information sharing is diegetically gated — rival factions don't share intelligence unless they physically share patrol space or have diplomatic contact. This allows the player to maintain trust with opposing factions simultaneously, as long as they manage visibility. A hidden InfoLeakTimer introduces risk if the player over-commits to two rivals: it opens a loyalty mission rather than immediately punishing.

CORE LOOP

EXPLORE	ENGAGE	ENCOUNTER	EARN	REPEAT
Move through world, talk to NPCs, gather intel, loot	Combat, puzzle, or social interaction triggered	Combat turns, dialogue, stealth, mini-games	Credits + Reputation / Damaged Familiar –	Push deeper, unlock shortcuts, uncover lore

STATUS & NEXT STEPS

- Drone flight prototype in UE5 Blueprints — targeting a playable vertical slice for IGDC 2026, Chennai (Oct 28–29)
- Koel Reactive Story System fully documented; faction disposition, NPC memory, and world state variable frameworks complete
- Full control scheme designed for PC, Xbox, and PS with UE5 Enhanced Input priority system
- World bible, narrative architecture, and pitch materials complete