**Phase 1 (Offline, Single-Player)**

**Scope & guardrails**

* **Emotions:** frustration, satisfaction
* **Hooks:** Outcome Streak, Progress Drought (social hook removed)
* **Session persistence:** reset all values at session start
* **Escalation Factor (EF):** 1.2
* **Rare+ boost cap:** ≤ **+20%** to computed delta (before clamping)
* **Soft cap per update:** ~85% of target peak; then clamp to [0, max]
* **Quiet window:** 6–8s global (prevents stackups)

**Hooks (only these 2)**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Hook** | **Condition** | **Affects** | **Base Δ** | **Scaling & cadence** | **Notes** |
| **Outcome Streak** | Streak ≥2 of similar outcomes | frustration / satisfaction | Common streak → **frustration +5**; Epic/Legendary → **satisfaction +2** | **Δ = base × EF^(streak−1)**; apply Rare+ boost ≤ +20% | Models “streak frustration” vs “reward spike”; soft-cap then clamp |
| **Progress Drought** | time\_since\_progress ≥ 60s **or** idle > 10 min | frustration | **+3** | Fires **every 10s** while active; scale with EF using drought\_streak | Reset drought\_streak when condition clears |

**Decay (event-conditioned, Phase 1 simple)**

* **frustration:** −1.5 when an **XP multiplier** or **dupe-conversion bonus** fires.
* **satisfaction:** −2.0 if **no Rare+ (epic/legendary/highlight)** for **two sessions**.  
  *(All sharing/leaderboard references removed.)*

**Update order (per event)**

1. Apply eligible **decay**
2. Evaluate **hooks**; compute deltas with **EF** scaling
3. If Rare+, apply boost **≤ +20%**
4. **Sum deltas per emotion**, **soft-cap (~85%)**, then **clamp**
5. Enforce **cooldowns/mutex** and **6–8s quiet window**

**Minimal JSON/state (per player, per session)**

{

"schema\_version": "ccas.p1.offline.v1",

"player\_id": "string",

"session\_id": "string",

"ts\_last\_update": 0,

"emotions": {

"frustration": { "value": 0.0 },

"satisfaction": { "value": 0.0 }

},

"streaks": { "success": 0, "fail": 0, "drought": 0 },

"config": {

"EF": 1.2,

"ct\_soft\_cap\_pct": 0.85,

"rare\_multiplier\_cap": 1.20,

"quiet\_window\_s": [6, 8]

}

}

**Supported event payloads**

{ "type": "outcome", "timestamp": 0,

"rarity": "common|uncommon|rare|epic|legendary" }

{ "type": "tick", "timestamp": 0,

"time\_since\_progress\_s": 0, "idle\_s": 0 }

{ "type": "meta", "timestamp": 0,

"xp\_multiplier\_fired": false, "dupe\_conversion\_bonus": false,

"session\_has\_rare\_plus": false }

**Acceptance checks**

* ✅ **3 commons in a row:** frustration += 5 × 1.2^(streak−1); per-update push ≤ ~85% of target.
* ✅ **Epic/Legendary:** satisfaction += 2 × 1.2^(streak−1); Rare+ extra ≤ **+20%** before clamp.
* ✅ **Drought active:** every 10s, frustration += 3 × 1.2^(drought\_streak−1); resets when progress resumes.
* ✅ **Decays:** frustration −1.5 on XP/dupe-bonus; satisfaction −2.0 if two sessions in a row have **no Rare+**.
* ✅ **Session reset** zeros emotions & streaks; **quiet window 6–8s** prevents multiple hook firings.