🔒 GOVERNANCE RECORD

₹0 Backend Migration — Cloud Run (Payments) with Resolver-Led Architecture

Status: FINAL · LOCKED · GOVERNED

Date Recorded: 2026-01-25

Applies To: Agile AI University — Portal, Entitlements, Executive Insight

---

✂️PURPOSE

This record documents the approved, governance-safe migration from Firebase Cloud Functions–based payment handling to a ₹0-cost backend using **Google Cloud Run (Free Tier)**, while **preserving** the resolver-led entitlement architecture, Firestore as the source of truth, and all locked frontend behavior.

This change is **cost-motivated**, **non-disruptive**, and **reversible**.

🖋️NON-NEGOTIABLES (LOCKED)

The following are **explicitly preserved**:

- Resolver-led architecture (no UI inference)
- Firestore as authoritative entitlement store
- Existing portal, dashboard, gating, and credential logic
- Firebase Auth usage and readiness contract
- Governance semantics for Executive Insight (paid > trial)

No frontend entitlement logic is modified.

🛏️REMOVALS (SCOPED & SAFE)

🔩Disabled in Firebase Cloud Functions

Only **payment-related exports** are disabled to eliminate billing risk:

- `createExecutiveOrder`
- `markExecutivePending`
- `razorpayWebhook`

These functions are **not deleted**, only disabled/commented.

🔗 Retained in Firebase Cloud Functions (LOW-COST)

The following remain active and governed:

- `resolvePortalEntitlements`
- `grantTrialEntitlementOnSignup`
- `syncExecutiveEntitlementToPortal`
- `checkExecutiveEntitlement`
- `corsSelfTest`

These functions are low-traffic and cost-safe.

🌂 ADDITIONS (NEW BACKEND)

Cloud Run Backend (₹0 Tier)

A new backend is introduced **outside** Firebase Functions:

```
/cloudrun-backend
  ├── index.js        # Express + Razorpay (payments only)
  ├── package.json    # Minimal deps
  ├── Dockerfile      # Cloud Run deploy
  └── .dockerignore
```

Responsibilities (STRICT): - Create Razorpay orders - Receive Razorpay webhooks - Write to Firestore (executiveEntitlements)

Explicitly NOT responsible for: - Entitlement resolution - UI logic - Dashboard access control

---

🌡️ FRONTEND CHANGE (SINGLE POINT)

Only one frontend change is permitted:

- Replace payment API URL

```
- /cloudfunctions.net/createExecutiveOrder
+ https://<cloud-run-url>/payments/create-order
```

No other frontend files are altered.

🎒 MANUAL OVERRIDE (REVENUE CONTINUITY)

To allow immediate revenue without automation dependency:

Manual entitlement grant is approved:

```
executiveEntitlements/{email}
{
  entitled: true
}
```

This is: - Resolver-compatible - Dashboard-safe - Audit-visible

Automation may follow post-revenue.

🎓BILLING & COST GUARANTEE

- Firebase Functions: minimal usage only
- Cloud Run: Free Tier (request-based)
- No prepaid balance required
- No Razorpay test payments required for dashboard testing

This architecture **unblocks revenue immediately** without ongoing platform cost anxiety.

🎩CHANGE AUTHORITY

- **Allowed:** Backend payment plumbing changes
- **Disallowed:** Resolver logic changes, UI gating changes, credential semantics changes

Any deviation requires explicit governance unlock.

🎽FINAL ASSERTION

This migration:

- Preserves institutional credibility
- Preserves architectural integrity
- Reduces operational risk
- Enables immediate monetization

Record Locked.