

Finanzas build forensics report

Finanzas SD – Architecture, Flows & SOPs

Arquitectura, Flujos y Procedimientos

January 22, 2026

[ARCHIVED] Superseded by docs/finanzas/overview.md on 2025-12-05 # Finanzas build forensics report

0.1 Symptoms

- Multiple deployments reported identical Finanzas bundles (index-*.js / index-*.css) and unchanged UI despite code edits.
- GitHub Actions currently lacks hash evidence, so it is unclear whether the build, S3 sync, or CloudFront caching is responsible.

0.2 Experiments and evidence

0.2.1 1) Baseline Finanzas build

- Command: `BUILD_TARGET=finanzas VITE_USE MOCKS=false VITE_API_BASE_URL=https:// npm run build`
- Result: build emitted `dist-finanzas/assets/index-CEiFFi4m.js` and `index-C2BIxzvu.css` with expected sizes. `973ea5†L1-L10`
- Hashes captured to `/tmp/baseline-hashes.txt` (JS and CSS). `1331a3†L1-L13`

0.2.2 2) Injected visible Finanzas change

- Edit: added TEST-FINZ-123 to the Finanzas home header (`src/modules/finanzas/FinanzasHome`)
- Rebuilt with the same command. Output JS asset changed to `index-DTjB0mGE.js`; CSS asset unchanged. `ace723†L1-L9`
- Hash diff: JS hash changed; CSS hash identical. `1331a3†L4-L13`
- The marker string is present inside the built Finanzas JS bundle, proving the change flowed into the Finanzas graph. `2da29f†L4-L6`

0.2.3 3) Cleanup

- Reverted the marker change and removed `dist-finanzas/` to keep the working tree clean (no functional code change left behind). `66abc0†L1-L2`

0.3 Findings

1. **Local Finanzas builds respond to UI changes.** A trivial header edit changed the JS filename and hash, and the marker appeared in the bundle. Therefore, Vite hashing is working and the Finanzas entry graph includes `FinanzasHome`.
2. **The earlier “frozen hash” observation is most consistent with CI deploying an unchanged build**, either because Finanzas-specific files were not modified in the commits being deployed or because CI redeployed an already-built

artifact without detecting staleness. The workflow currently provides no visibility into bundle hashes, so this scenario would go unnoticed.

0.4 Validated root cause

- There is no evidence that Vite is misconfigured; instead, the pipeline lacks safeguards and observability. Without hash logging or comparison, CI can ship the previous Finanzas bundle even when the code changes (or when edits land outside the Finanzas entry graph), leading to the perception of a “stuck” bundle.

0.5 Remediation plan

1. **Add an explicit hash check script** (`scripts/finanzas-build-hash-check.sh`) to standardize local verification and optional comparisons against a reference hash file.
2. **Instrument the Deploy Finanzas UI workflow** to:
 - Record JS/CSS hashes in `$GITHUB_STEP_SUMMARY` and upload them as an artifact.
 - Download the last successful run’s hash artifact and fail the deployment if hashes are unchanged while Finanzas-facing files changed in the current commit.
3. **Use the checklist below for future diagnostics:**
 1. Run `scripts/finanzas-build-hash-check.sh` (optionally with a reference hash file) to confirm local bundle deltas.
 2. Inspect the GitHub Actions step summary for the recorded hashes.
 3. Compare the recorded hashes with the prior successful run; investigate if hashes repeat while Finanzas files changed.
 4. If hashes differ but UI looks stale, proceed to the existing S3 manifest and CloudFront smoke steps to isolate caching vs. build issues.