Finanzas SDMT Changes API – 500s investigation

Finanzas SD – Architecture, Flows & SOPs

Arquitectura, Flujos y Procedimientos

December 15, 2025

1 Finanzas SDMT Changes API – 500s investigation

1.1 Observations

- **Infrastructure wiring**: The SAM template exposes GET/POST `/projects/{projectId}/chang`
  on the ChangesFn Lambda, which is built from `src/handlers/changes.ts`, and
  grants it CRUD access to the ChangesTable (`${TablePrefix}changes` with pk/sk
  keys).〖F:services/finanzas-api/template.yaml†L1220-L1258〗〖F:services/finanzas-
  api/template.yaml†L214-L231〗

- **Handler behavior**:

  - GET validates `projectId`, calls ensureCanRead, then queries DynamoDB for
    `pk = PROJECT#${projectId}` and returns `{ data, projectId, total }` af-
    ter normalizing items.〖F:services/finanzas-api/src/handlers/changes.ts†L76-
    L188〗
  - POST validates JSON, enforces `title`, `description`, and numeric `impact_amount`,
    builds a `CHANGE#` item keyed by `PROJECT#${projectId}`, and writes with
    a ConditionExpression to prevent duplicates. Auth (ensureCanWrite +
    getUserEmail) runs before DynamoDB writes. Failures fall through to the
    catch block, which returns 500 for any non-auth errors.〖F:services/finanzas-
    api/src/handlers/changes.ts†L91-L200〗

- **Contract test expectations**: The dev contract script first selects a project from
  `/projects`, then issues GET `/projects/{id}/changes` and expects 2xx. It POSTs
  a payload containing `title`, `description`, numeric `impact_amount`, `currency`,
  `justification`, and an array `affected_line_items`, then re-GETs and asserts
  the new id is present; any non-2xx is treated as a failure.〖F:tests/finanzas/changes/run-
  changes-tests.sh†L37-L123〗

- **Frontend payloads**:

  - getChangeRequests fetches `/projects/${project_id}/changes`, accept-
    ing multiple payload shapes and normalizing to the domain model, defaulting
    status to pending if missing.〖F:src/lib/api.ts†L755-L769〗
  - createChangeRequest POSTs `{ baseline_id, title, description, impact_amount,`
    `currency, justification, affected_line_items }` to the same route and
    normalizes the response; it doesn't send `project_id` in the body, relying on
    the path parameter.〖F:src/lib/api.ts†L772-L818〗
  - The SDMT Changes form splits the "Affected line items" textbox into an array
    and passes the same fields (`title`, `description`, `impact_amount`, `currency`,
    `baseline_id`, `justification`, `affected_line_items`) to createChangeRequest.
    Project id comes from ProjectContext selection, and missing/invalid values
    are blocked client-side.〖F:src/features/sdmt/cost/Changes/SDMTChanges.tsx†L71-

L152⟧

1.2 Likely 500 triggers in dev

1. **DynamoDB resource mismatch**: The handler unconditionally calls `tableName("changes")` for both GET and POST. If the deployed stack is missing the `${TablePrefix}changes` table or TABLE_CHANGES is misconfigured, `QueryCommand`/`PutCommand` will throw (ResourceNotFound/AccessDenied), leading to the generic `serverError` path (HTTP 500). This is consistent with 500s on both load and create despite green unit/contract tests, which rely on a properly provisioned table. The template declares the table, but a drifted or partial dev deployment would surface exactly as the reported 500s.⟦F:services/finanzas-api/src/handlers/changes.ts†L76-L200⟧⟦F:services/finanzas-api/template.yaml†L214-L231⟧

2. **Unhandled Dynamo conditional failures**: POST uses a strict `ConditionExpression` prohibiting any existing pk/sk; if the dev data layer reuses change IDs (e.g., seeded CHANGE# ids or retries with the same id), Dynamo returns `ConditionalCheckFailedExce` which also bubbles to the 500 catch block. Contract tests generate unique ids and won't catch this, but a stuck id in dev would trigger 500 on create.⟦F:services/finanzas-api/src/handlers/changes.ts†L134-L168⟧

3. **Environment/audience drift**: Auth errors are converted to 4xx by `fromAuthError`, but any other runtime exceptions from token parsing or missing env (e.g., `COGNITO_CLIENT_ID`) would throw before reaching the handler catch. Given the handler defers to `ensureCanRead/Write`, a misconfigured dev client id could surface as 500s even though tests inject mocks. Confirming Cognito env matches the UI client id is necessary.⟦F:services/finanzas-api/src/handlers/changes.ts†L181-L199⟧⟦F:services/finanzas-api/src/lib/auth.ts†L33-L89⟧

1.3 Recommended fixes (surgical)

1. **Harden Dynamo wiring & error surfaces**

   - Add explicit handling for `ResourceNotFoundException` and `ConditionalCheckFailedExce` in `changes.ts` so they return actionable 4xx/409 responses instead of 500, and log the resolved table name for visibility. This will convert missing-table drift into a clear signal and prevent UI 500s on duplicate ids.⟦F:services/finanzas-api/src/handlers/changes.ts†L76-L200⟧
   - Consider guarding `tableName("changes")` with a startup validation (or emitting the resolved name in logs) to detect env drift early.

2. **Verify dev stack resources**

   - Confirm `${TablePrefix}changes` exists and matches the Lambda env (TABLE_CHANGES), and that ChangesFn has CRUD permissions. If absent, redeploy the ChangesTable

resource from `template.yaml` or correct the prefix for the dev stage.【F:services/finanzas-api/template.yaml†L214-L231】【F:services/finanzas-api/template.yaml†L1220-L1258】

3. **Protect against duplicate ids**

- Either ensure frontend never supplies `id` (current UI doesn't) or make the handler generate the id unconditionally when `sk` already exists, returning 409 on collision. This prevents silent retries from surfacing as 500s.【F:services/finanzas-api/src/handlers/changes.ts†L134-L168】

1.4 Test/validation plan

- Backend: `cd services/finanzas-api && npm test --runInBand`.
- Contract: `bash tests/finanzas/changes/run-changes-tests.sh` and `bash tests/finanza`
- Frontend build sanity: `npm run build:finanzas`.
- UI: In `/finanzas/sdmt/cost/changes`, pick a known project (e.g., the first from `/projects`), load the page (expect no 500s), submit a new change with title/description/impact, and confirm it appears in the table and summary counts with status pending.