

# **UI-API Action Map**

Finanzas SD – Architecture, Flows & SOPs

Arquitectura, Flujos y Procedimientos

January 18, 2026

## 1 UI-API Action Map

This document maps UI actions (buttons, controls, forms) to their corresponding API client methods and HTTP endpoints.

### 1.1 Forecast Management

#### 1.1.1 PMO Forecast Adjustment (SDMT Forecast Page)

**UI Action:** “Guardar Pronóstico” button (PMO only) - **Location:** src/features/sdmt/cost/Forecast - **Client Method:** finanzasClient.bulkUpsertForecast(projectId, items) - **HTTP Endpoint:** PUT /projects/{projectId}/allocations:bulk?type=forecast - **Auth Required:** PMO role - **Request Body:** json { "items": [ { "rubroId": "MOD-ING", "month": "2025-01", "forecast": 12000 } ] } - **Response:** json { "success": true, "updated": 10, "skipped": 0, "total": 10 } - **UI Effect:** - Toast notification on success/error - Invalidate React Query keys for forecast data - Clear pending forecast edits state - Refresh KPI cards to show updated forecast totals

### 1.2 Budget Management

#### 1.2.1 Annual All-In Budget (Hub de Desempeño)

**UI Action:** Year selector change - **Location:** src/modules/finanzas/HubDesempeno.tsx (Budget section) - **Client Methods:** - finanzasClient.getAllInBudget(year) - finanzasClient. - **HTTP Endpoints:** - GET /budgets/all-in?year={year} - GET /budgets/all-in/overview?year={year} - **Auth Required:** Any authenticated user (SDMT, EXEC\_RO, PMO) - **UI Effect:** - Load budget amount for selected year - Load cross-project totals and variances - Update KPI tiles with budget metrics

**UI Action:** “Guardar Presupuesto” button (PMO only) - **Location:** src/modules/finanzas/HubDesempeno.tsx (Budget section) - **Client Method:** finanzasClient.putAllInBudget(year, amount, currency) - **HTTP Endpoint:** PUT /budgets/all-in?year={year} - **Auth Required:** PMO or ADMIN role - **Request Body:** json { "amount": 5000000, "currency": "USD" } - **Response:** json { "year": 2026, "amount": 5000000, "currency": "USD", "updated\_at": "2024-12-17T00:00:00Z", "updated\_by": "pmo@example.com" } - **UI Effect:** - Toast notification on success/error - Refresh budget overview to show updated variances - Update percentage consumed metrics

### 1.3 Existing Actions (Reference)

#### 1.3.1 Project Rubros Assignment

- **UI:** Add rubro button in project settings

- **Client Method:** `finanzasClient.createProjectRubro(projectId, payload)`
- **HTTP Endpoint:** `POST /projects/{projectId}/rubros`

### 1.3.2 Bulk Allocations Update

- **UI:** Save allocations button
- **Client Method:** `finanzasClient.saveAllocations(projectId, payload)`
- **HTTP Endpoint:** `PUT /projects/{projectId}/allocations:bulk`
- **Note:** This is for planned allocations (SDMT), not forecast adjustments

### 1.3.3 Provider Creation

- **UI:** Create provider form
- **Client Method:** `finanzasClient.createProvider(payload)`
- **HTTP Endpoint:** `POST /providers`

## 1.4 Notes

- All endpoints require authentication via Cognito JWT token
- PMO-specific actions will return 403 Forbidden if attempted by non-PMO users
- React Query keys should be invalidated after successful mutations to ensure UI consistency
- Error handling should provide user-friendly toast messages, not raw API errors