

# **ADR-0002: Separate API Gateway for Finanzas SD Domain**

Finanzas SD – Architecture, Flows & SOPs

Arquitectura, Flujos y Procedimientos

November 11, 2025

## 1 ADR-0002: Separate API Gateway for Finanzas SD Domain

**Status:** Accepted

**Date:** 2025-11-02

**Context:** Financial Planning-U

---

### 1.1 Decision

We will maintain a **dedicated API Gateway** (finanzas-sd-api) for the Finanzas Service Delivery (SD) business domain, separate from other modules such as Project Management (PM) or future domains.

### 1.2 Context

The Financial Planning-U platform is architected as a modular system where different business domains have distinct:

- **Lifecycles:** Different release cadences and deployment schedules
- **Security requirements:** Domain-specific RBAC, authentication scopes, and compliance needs
- **Scaling characteristics:** Varying traffic patterns, quotas, and performance requirements
- **Team ownership:** Independent teams managing their respective domains

The Finanzas SD module specifically handles:  
- Financial project planning and tracking  
- Budget allocation and management  
- Payroll integration  
- Service delivery control  
- Provider management  
- Budget variance alerts (PEP-3)

This module requires:  
- **Cognito JWT authentication** with SDT (Service Delivery Team) group validation  
- **Independent deployment** cycles without impacting other domains  
- **Fine-grained observability** for financial operations  
- **Strict rate limiting** and quotas for financial data operations

### 1.3 Alternatives Considered

#### 1.3.1 Option 1: Shared API Gateway with Path-Based Routing

**Pros:** - Single infrastructure component to manage  
- Potentially lower AWS costs (fewer API Gateways)  
- Unified API domain

**Cons:** - **Blast radius:** Deployment or configuration changes in one domain affect all domains  
- **RBAC complexity:** Mixing authorization rules across domains in one gateway  
- **Scaling constraints:** Single throttling/quota pool shared across unrelated

workloads - **Observability**: Difficult to isolate metrics, logs, and dashboards per domain - **Release coupling**: Changes to one domain require coordination with others

### 1.3.2 Option 2: Separate API Gateways per Domain (Selected)

**Pros:** - **Blast-radius isolation**: SD changes/rollouts don't risk PM or other areas - **Security & RBAC**: Dedicated Cognito app client + SDT group checks, per-API IAM policies - **Independent scaling & quotas**: Per-API throttles, caching, alarms, and budgets - **Clear observability**: Separate logs/metrics/dashboards; easier SLO/SLA tracking - **Lifecycle independence**: SD module can evolve schema and rate limits autonomously - **Cost visibility**: Better tagging and cost allocation per domain

**Cons:** - Slight increase in infrastructure complexity (multiple API Gateways) - Minor cost increase (though minimal for HTTP APIs in PAY\_PER\_REQUEST mode)

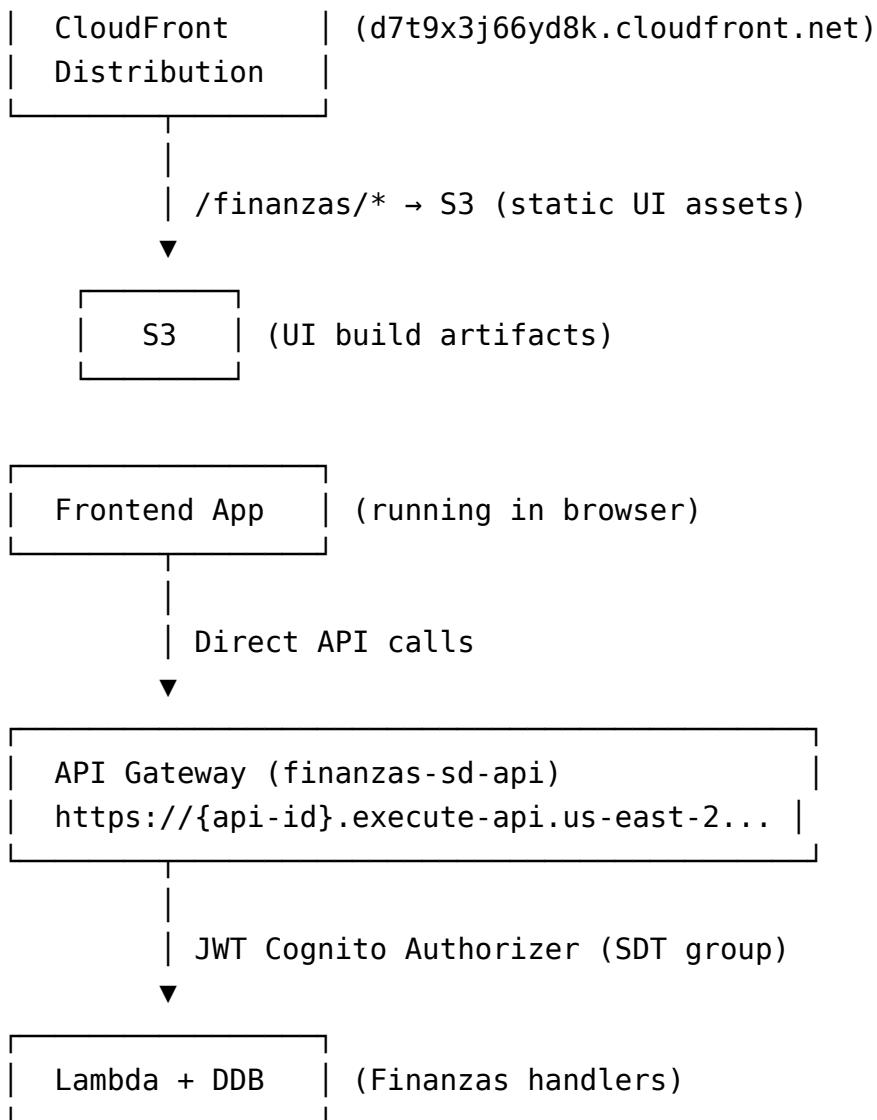
## 1.4 Decision Rationale

We choose **Option 2** (separate API Gateway) because:

1. **Operational safety**: Financial operations are critical. Isolating the API Gateway ensures that non-financial deployments cannot accidentally impact financial services.
2. **Security posture**: Financial data requires strict access controls. A dedicated gateway allows us to:
  - Enforce SDT group membership at the API level
  - Apply custom WAF rules specific to financial operations
  - Maintain separate IAM policies and resource-based policies
  - Implement domain-specific rate limiting for financial endpoints
3. **Scalability**: Financial operations may have different traffic patterns (e.g., end-of-month spikes for payroll). Independent quotas prevent resource contention.
4. **Compliance & audit**: Financial systems often require detailed audit trails. Separate CloudWatch log groups and X-Ray traces simplify compliance reporting.
5. **Team velocity**: The SD team can deploy, test, and rollback independently without coordinating with other teams.

## 1.5 Implementation

### 1.5.1 Current Architecture (Mode A)



### 1.5.2 Components

1. **CloudFront Distribution:** Serves static UI assets at /finanzas/\* from S3
2. **API Gateway (finanzas-sd-api):**
  - Dedicated HTTP API in us-east-2
  - Cognito JWT authorizer (issuer: User Pool us-east-2\_FyHLt0hiY)
  - CORS configured for CloudFront domain
3. **Cognito User Pool:** Shared with other modules, but dedicated app client for Finanzas SD
4. **Lambda Functions:** Finanzas business logic (projects, allocations, payroll, etc.)
5. **DynamoDB Tables:** Finanzas data stores (projects, rubros, allocations, etc.)

### 1.5.3 CORS Configuration

```
[ ] CorsConfiguration: AllowOrigins: -https://d7t9x3j66yd8k.cloudfront.net AllowMethods:[GET,POST,PUT,OPTIONS] AllowHeaders:['Authorization','Content-Type'] MaxAge:3600
```

### 1.5.4 Optional Enhancement: Mode B (CloudFront API Proxy)

Future consideration: Proxy API requests through CloudFront at /api/finanzas/\* for:

- Single origin domain for UI and API - Potential latency reduction via CloudFront edge locations
- Unified domain for simplified CORS

**Requirements for Mode B:** - Add API Gateway as CloudFront origin - Configure behavior for /api/finanzas/\* path pattern - Use origin request policy to forward Authorization header and query strings - Update CORS to allow CloudFront domain or use '\*' for development

**Current Status:** Mode B is documented as a **backlog item** for future implementation if needed.

## 1.6 Consequences

### 1.6.1 Positive

- ☐ Independent deployment and rollback for Finanzas SD
- ☐ Isolated failure domains (API issues don't cascade)
- ☐ Clear security boundaries and audit trails
- ☐ Flexible rate limiting and quota management
- ☐ Improved observability and debugging

### 1.6.2 Negative

- Additional infrastructure component to manage
- Slight increase in AWS costs (minimal for HTTP API)
- Need to manage multiple API Gateway configurations

### 1.6.3 Neutral

- Frontend must be configured with the correct API base URL
- OpenAPI specification uses variableized servers section

## 1.7 References

- SAM Template
- Finanzas Architecture
- API Gateway HTTP APIs Documentation

- Deploy Workflow

## 1.8 Related Decisions

- ADR-0001: (Future) OIDC authentication for CI/CD
  - ADR-0003: (Future) DynamoDB table design for Finanzas SD
- 

**Review:** To be reviewed quarterly or when scaling requirements change significantly.