

# **Finanzas SD - AWS Architecture Diagrams**

Finanzas SD – Architecture, Flows & SOPs

Arquitectura, Flujos y Procedimientos

November 11, 2025

## 1 Finanzas SD - AWS Architecture Diagrams

This directory contains enterprise-grade AWS architecture diagrams for the Finanzas SD (Service Delivery Financial Planning) module. All diagrams follow AWS 2025 architecture framework standards with professional styling and comprehensive documentation.

### 1.1 Diagram Inventory

#### 1.1.1 1. AWS Architecture Overview (01-aws-architecture-overview.mmd)

**Purpose:** High-level system architecture showing all AWS services and their interactions

**Style:** Horizontal AWS web application hosting layout

**Components:** - CloudFront CDN with /finanzas/\* path pattern - Cognito User Pool (JWT authentication, RBAC groups) - API Gateway REST with Verified Permissions - Lambda Functions (PMO, SDMT, Forecast, Analytics, Reconciliation) - DynamoDB Tables (Projects, Budgets, Forecasts, Invoices) - S3 Buckets (UI assets, documents) - EventBridge scheduled jobs - SNS/SES notifications - CloudWatch & X-Ray observability

**Export Formats:** PNG, SVG, PDF

---

#### 1.1.2 2. CI/CD Pipeline (02-cicd-pipeline.mmd)

**Purpose:** Multi-stage deployment pipeline with quality gates

**Style:** AWS CI/CD with GitHub Actions integration

**Components:** - GitHub Actions workflows (frontend, backend, docs) - Lint, test, build, security scan stages - OIDC authentication to AWS - Multi-environment deployment (dev, staging, prod) - SAM deploy for backend - S3 sync + CloudFront invalidation for frontend - Smoke tests and monitoring - Rollback strategy

**Export Formats:** PNG, SVG, PDF

---

#### 1.1.3 3. Data Lifecycle & Analytics (03-data-lifecycle-analytics.mmd)

**Purpose:** Time-series data processing for forecasting and cash flow analysis

**Style:** AWS time series analytics pipeline

**Components:** - Data ingestion (API Gateway, S3 raw bucket) - ETL processing (Lambda functions) - DynamoDB time-series storage - Analytics Lambda (cash flow, variance,

reporting) - S3 export bucket - SharePoint integration - EventBridge scheduled jobs - CloudWatch metrics & X-Ray tracing

**Export Formats:** PNG, SVG, PDF

---

#### 1.1.4 4. Business Process Flow (04-business-process-flow.mmd)

**Purpose:** End-to-end finance operations workflow

**Style:** Business process flow / BPMN hybrid

**Phases:** 1. **Planning & Estimation** - PMO project initiation, budget estimator 2. **Budget Baseline Creation** - Service tier selection, digital signature 3. **Forecast Allocation** - 60-month grid, period-by-period allocation 4. **Execution & Tracking** - Actuals recording, invoice receipt 5. **Invoice Reconciliation** - Automated matching, ML algorithm 6. **Analytics & Reporting** - Cash flow, variance, margin analysis 7. **Approval & Governance** - Alerts, approval workflow, audit trail 8. **Document Generation** - PDF/Excel export, SharePoint upload

**Export Formats:** PNG, SVG, PDF

---

#### 1.1.5 5. Network & Security Architecture (05-network-security.mmd)

**Purpose:** Security architecture and network connectivity

**Style:** AWS network diagram with security layers

**Components:** - Route 53 DNS - AWS WAF + Shield (DDoS protection) - CloudFront with SSL/TLS - S3 with Origin Access Control (OAC) - Cognito + Verified Permissions (authentication & authorization) - API Gateway with custom authorizer - Lambda IAM roles and execution policies - DynamoDB & S3 encryption (KMS) - Secrets Manager (credentials rotation) - CloudTrail (audit logs) - CloudWatch & X-Ray (monitoring)

**Export Formats:** PNG, SVG, PDF

---

## 1.2 □ Design Standards

### 1.2.1 Color Coding

All diagrams follow AWS 2025 color palette:

Color	Hex Code	Usage
<b>AWS Orange</b>	#FF9900	CDN, Storage, Frontend

Color	Hex Code	Usage
<b>AWS Blue</b>	#146EB4	API Gateway, Auth, Networking
<b>AWS Purple</b>	#8B5CF6	Compute (Lambda), Processing
<b>AWS Green</b>	#3F8624	Data Layer (DynamoDB, Analytics)
<b>AWS Red</b>	#D13212	Security, Monitoring, Alerts
<b>AWS Dark</b>	#232F3E	Borders, Text
<b>AWS Gray</b>	#545B64	Connection Lines

### 1.2.2 Connection Arrow Meanings

- **Solid arrows** (- ->) - Primary data flow or request path
- **Dotted arrows** (- . ->) - Secondary flow, monitoring, or event-driven
- **Labeled arrows** - Numbered steps or action descriptions

## 1.3 Generating Diagrams

### 1.3.1 Prerequisites

```
[ ] npm install -g @mermaid-js/mermaid-cli@11.4.1
```

### 1.3.2 Generate All Diagrams

```
[ ] # From repository root ./scripts/docs/render-docs.sh
```

This will: 1. Validate Mermaid syntax 2. Render diagrams to SVG, PNG, and PDF 3. Output to public/docs/latest/diagrams/

### 1.3.3 Generate Single Diagram

```
[ ] # SVG output mmdc -i docs/diagrams/01-aws-architecture-overview.mmd -o out-
put.svg -t base
# PNG output (high resolution) mmdc -i docs/diagrams/01-aws-architecture-overview.mmd
-o output.png -t base -w 2400 -H 1600
# PDF output mmdc -i docs/diagrams/01-aws-architecture-overview.mmd -o out-
put.pdf -t base
```

## 1.4 □ Export Locations

Generated diagrams are exported to multiple locations:

1. **Public Documentation:** `public/docs/latest/diagrams/`
    - Included in documentation website
    - Accessible via CloudFront at `/docs/latest/`
  2. **GitHub Actions Artifacts:**
    - Workflow: `.github/workflows/docs-generator.yml`
    - Retention: 90 days
    - Download from Actions tab
  3. **Client Deliverables:** `public/docs/releases/`
    - Packaged in branded ZIP files
    - Includes bilingual documentation
- 

## 1.5 □ Automatic Updates

Diagrams are automatically regenerated when:

1. Any `.mmd` file in `docs/diagrams/` or `diagrams/` is modified
2. Documentation generation workflow is manually triggered
3. A new release is created

Workflow: `.github/workflows/docs-generator.yml`

---

## 1.6 □ Maintenance

### 1.6.1 Updating Diagrams

1. Edit the `.mmd` file using Mermaid syntax
2. Validate syntax: `./scripts/docs/validate-diagrams.sh`
3. Commit changes to feature branch
4. Open PR - diagrams will be rendered in CI/CD
5. Review rendered output in PR artifacts

### 1.6.2 Adding New Diagrams

1. Create `.mmd` file in `docs/diagrams/` with sequential naming: `06-diagram-name.mmd`
2. Include frontmatter with title and description
3. Follow existing color scheme and styling
4. Update this README with diagram details
5. Commit and open PR

### 1.6.3 Best Practices

- **Keep diagrams focused:** One diagram per architectural concern
  - **Use consistent naming:** Service names should match AWS SAM template
  - **Label connections:** All arrows should have descriptions
  - **Include legends:** When introducing new symbols or colors
  - **Test rendering:** Always validate before committing
- 

## 1.7 References

### 1.7.1 Mermaid Documentation

- Mermaid Official Docs
- Flowchart Syntax
- Sequence Diagrams
- Theming




### 1.7.2 AWS Architecture Icons

- AWS Architecture Icons
- AWS Architecture Center
- AWS Well-Architected Framework

### 1.7.3 Related Documentation

- AWS\_Architecture.md - Technical architecture details
  - Data\_Flows.md - Data flow descriptions
  - deploy.md - Deployment procedures
  - DEPLOYMENT\_GUIDE.md - Full deployment guide
- 

## 1.8 Security Considerations

These diagrams are **internal documentation** and should be treated accordingly: -  
 Safe to share with internal teams and stakeholders -  Include in client deliverables (sanitized) -  Redact sensitive information before external sharing: - Account IDs - API endpoint URLs - CloudFront distribution IDs - S3 bucket names - Cognito Pool IDs

---

## 1.9 Version History

---

Version	Date	Changes	Author
1.0	2025-11-10	Initial diagram set (5 diagrams)	Copilot

---

## 1.10 Future Enhancements

Planned additions: - ☐ Cost optimization diagram (Reserved Capacity, Savings Plans) - ☐ Disaster recovery architecture - ☐ Multi-region deployment (if needed) - ☐ Integration architecture with external systems - ☐ Database schema ERD with relationships - ☐ User journey flow diagrams

---

## 1.11 Support

For questions or issues with diagrams: 1. Check DOCUMENTATION\_PIPELINE.md 2. Review DOCS\_PIPELINE\_SUMMARY.md 3. Open an issue in GitHub with label documentation

---

**Last Updated:** November 10, 2025

**Maintained By:** Platform Team

**Repository:** valencia94/financial-planning-u