**Họ và tên: Đinh Hoàng Thanh Phong**

**Trường: Đại học công nghệ TP.HCM(HUTECH)**

**Đề tài:**

**Triển khai hệ thống quản lý ngân sách thông minh với cảnh báo dự đoán, kiểm soát tự động, và phân tích phương sai trên AWS**

**Mục lục**

[1. Problem Statement 5](#_Toc203055120)

[1.1 Current Situation Analysis 5](#_Toc203055121)

[1.2 Pain Points Identification with Quantified Impact 6](#_Toc203055122)

[1.3 Stakeholders Affected and Their Concerns 6](#_Toc203055123)

[1.4 Business Consequences of Inaction 7](#_Toc203055124)

[1.5 Market Opportunity 7](#_Toc203055125)

[2. Solution Architecture 8](#_Toc203055126)

[2.1 Architecture Overview 8](#_Toc203055127)

[2.2 AWS Services Selection and Justification 9](#_Toc203055128)

[2.3 Component Interactions and Data Flow 10](#_Toc203055129)

[2.4 Security Architecture and Compliance 11](#_Toc203055130)

[2.5 Scalability and Performance Considerations 11](#_Toc203055131)

[2.6 Integration Points with Existing Systems 12](#_Toc203055132)

[3. Technical Implementation 12](#_Toc203055133)

[3.1 Implementation Phases with Deliverables 12](#_Toc203055134)

[3.2 Technical Requirements 13](#_Toc203055135)

[3.3 Development Approach and Methodologies 14](#_Toc203055136)

[3.4 Testing Strategy 14](#_Toc203055137)

[3.5 Deployment Plan and Rollback Procedures 15](#_Toc203055138)

[3.6 Configuration Management 16](#_Toc203055139)

[4. Timeline & Milestones 16](#_Toc203055140)

[4.1 Project Phases Breakdown 16](#_Toc203055141)

[4.2 Key Milestones with Success Criteria 17](#_Toc203055142)

[4.3 Dependencies Identification 17](#_Toc203055143)

[4.4 Critical Path Analysis 18](#_Toc203055144)

[4.5 Resource Allocation Plan 18](#_Toc203055145)

[4.6 Buffer Time for Risks 19](#_Toc203055146)

[5. Budget Estimation 19](#_Toc203055147)

[5.1 Overview 19](#_Toc203055148)

[5.2 AWS Infrastructure Costs 20](#_Toc203055149)

[5.3 Development Costs (One-time) 20](#_Toc203055150)

[5.4 Third-party Services and Licenses 21](#_Toc203055151)

[5.5 Operational Costs (Ongoing) 21](#_Toc203055152)

[5.6 ROI Calculation and Break-even Analysis 21](#_Toc203055153)

[5.7 Cost Optimization Strategies 22](#_Toc203055154)

[6. Risk Assessment 23](#_Toc203055155)

[6.1 Overview 23](#_Toc203055156)

[6.2 Risk Identification 23](#_Toc203055157)

[**6.2.1 Technical Risks** 23](#_Toc203055158)

[**6.2.2 Business Risks** 23](#_Toc203055159)

[**6.2.3 Operational Risks** 23](#_Toc203055160)

[6.3 Impact Assessment and Probability Analysis 24](#_Toc203055161)

[6.4 Risk Matrix with Prioritization 24](#_Toc203055162)

[6.5 Mitigation Strategies for Each Risk 25](#_Toc203055163)

[6.6 Contingency Plans 25](#_Toc203055164)

[6.7 Monitoring and Escalation Procedures 26](#_Toc203055165)

[7. Expected Outcomes 26](#_Toc203055166)

[7.1 Overview 26](#_Toc203055167)

[7.2 Success Metrics 27](#_Toc203055168)

[7.3 Short-term Benefits (0–6 months) 27](#_Toc203055169)

[7.4 Medium-term Benefits (6–18 months) 27](#_Toc203055170)

[7.5 Long-term Value (18+ months) 28](#_Toc203055171)

[7.6 User Experience Improvements 28](#_Toc203055172)

[7.7 Strategic Capabilities Gained 29](#_Toc203055173)

[Appendix A. Technical Specifications 29](#_Toc203055174)

[AWS Services and Configurations 29](#_Toc203055175)

[IAM Roles and Policies 30](#_Toc203055176)

[Appendix D. References 32](#_Toc203055177)

**Executive Summary**

**1. Purpose and Context**

Chi phí vận hành công nghệ, đặc biệt là dịch vụ đám mây, ngày càng trở thành một phần lớn trong ngân sách doanh nghiệp. Tuy nhiên, nhiều tổ chức vẫn dựa vào quy trình quản lý ngân sách thủ công hoặc phân tán – dẫn đến thiếu cảnh báo kịp thời, kiểm soát yếu, và phát sinh chi phí vượt kế hoạch.

Giải pháp này được đề xuất để giải quyết vấn đề đó thông qua triển khai một hệ thống quản lý ngân sách tự động (Automated Budget Management) tích hợp các tính năng:

* Cảnh báo dự đoán (predictive alerting) dựa trên forecast
* Cơ chế kiểm soát tự động khi chạm ngưỡng
* Phân tích phương sai (variance analysis) giữa kế hoạch và thực tế
* Dashboard báo cáo chi tiết, real-time

Mục tiêu của dự án là giúp doanh nghiệp giảm rủi ro overspend, tăng tính minh bạch và hiệu quả trong quản trị tài chính cho hạ tầng công nghệ.

**2. Problem Statement (Short Version)**

Hiện tại, việc quản lý ngân sách thường dựa vào:

* Báo cáo hậu kiểm, trễ so với phát sinh thực tế
* Quy trình phê duyệt và kiểm soát thủ công
* Thiếu cơ chế cảnh báo dự đoán khi chi phí vượt ngưỡng

Hậu quả chính:

* Overspend trung bình 10–30%
* Thiếu dữ liệu để ra quyết định nhanh
* Tăng rủi ro kiểm toán và compliance

Việc không cải thiện sẽ dẫn đến chi phí bị lãng phí, giảm lợi nhuận và mất khả năng cạnh tranh.

**3. Solution Overview with Key Features**

Giải pháp đề xuất là một hệ thống quản lý ngân sách thông minh trên AWS với các thành phần chính:

**AWS Budgets with Predictive Alerts**

* Định nghĩa ngân sách và ngưỡng cảnh báo
* Cảnh báo khi dự báo (forecast) vượt giới hạn

**Automation & Controls**

* Lambda function tự động xử lý sự kiện
* Có thể khóa, giới hạn hoặc gửi phê duyệt khi overspend

**Variance Analysis Module**

* Thu thập dữ liệu chi phí thực tế (Cost Explorer)
* So sánh với kế hoạch theo cost center, service, project

**Dashboard and Reporting**

* QuickSight Dashboard hiển thị variance, forecast, alert history
* Phân quyền truy cập theo vai trò (Finance, IT, Execs)

**Escalation Workflow**

* Quy trình thông báo và phê duyệt đa cấp
* Tích hợp email/SNS/EventBridge

**4. Business Benefits and ROI Summary**

Giảm rủi ro chi tiêu vượt ngân sách tới 30%  
 Tăng minh bạch và kiểm soát cho Finance và IT  
 Tối ưu quy trình phê duyệt, giảm công sức thủ công  
 Hỗ trợ kiểm toán và tuân thủ (compliance) tốt hơn  
 ROI kỳ vọng đạt mức hoàn vốn sau 12–18 tháng  
 Tiềm năng mở rộng đa phòng ban và toàn doanh nghiệp

**5. Investment Required and Timeline**

**Chi phí đầu tư ban đầu (ước tính):**

* Phát triển (nhân lực nội bộ / outsource): ~1,000–1,500 USD
* Hạ tầng AWS (serverless): ~50–100 USD/tháng vận hành

**Timeline triển khai (dự kiến):**

* Tuần 1: Thu thập yêu cầu, thiết kế kiến trúc
* Tuần 2: Cài đặt AWS Budgets, SNS, Lambda
* Tuần 3: Phát triển Variance Analysis Module
* Tuần 4: Hoàn thiện Dashboard, UAT, Handover

**Phạm vi:** Proof-of-Concept (PoC) 3–4 tuần với khả năng mở rộng sau.

**6. Success Metrics and Expected Outcomes**

**Success Metrics:**

* 95% cảnh báo đúng hạn (alert accuracy)
* Giảm variance trung bình 20% trong 6 tháng
* Adoption rate >80% cho bộ phận Tài chính/IT

**Expected Outcomes:**

* Kiểm soát chi phí chủ động, realtime
* Giảm sự phụ thuộc quy trình thủ công
* Tăng cường tuân thủ chính sách tài chính
* Hỗ trợ chuyển đổi FinOps trong doanh nghiệp

**7. Conclusion**

Giải pháp Automated Budget Management with Predictive Alerting and Controls mang lại một nền tảng quản trị ngân sách thông minh, dễ triển khai trên AWS với chi phí thấp và hiệu quả cao.

Nó hỗ trợ doanh nghiệp chủ động giám sát, kiểm soát và tối ưu chi phí vận hành công nghệ, đồng thời nâng cao khả năng ra quyết định chiến lược dựa trên dữ liệu realtime và minh bạch.

# 1. Problem Statement

## 1.1 Current Situation Analysis

Hiện tại, hoạt động quản lý ngân sách công nghệ (đặc biệt là ngân sách hạ tầng cloud) trong doanh nghiệp chủ yếu dựa vào quy trình thủ công và các công cụ rời rạc như Excel, Email và báo cáo định kỳ.

Quy trình thông thường:

* Tạo kế hoạch ngân sách theo quý/năm trên bảng tính
* Theo dõi chi phí thực tế thông qua báo cáo của nhà cung cấp (AWS Cost Explorer, billing PDF)
* So sánh ngân sách và thực tế sau khi chi phí đã phát sinh
* Trao đổi phê duyệt và ra quyết định cắt giảm/thay đổi thủ công

**Hạn chế chính:**

* Thiếu khả năng giám sát realtime
* Không có cảnh báo trước khi vượt ngưỡng
* Phản ứng chậm với biến động chi phí
* Dữ liệu phân mảnh, thiếu tích hợp giữa IT và Finance

Kết quả là chi phí vận hành thường bị vượt ngân sách mà không được phát hiện kịp thời.

## 1.2 Pain Points Identification with Quantified Impact

Dựa trên khảo sát nội bộ và tham chiếu từ báo cáo FinOps Foundation:

*Thiếu cảnh báo kịp thời:*

* 65% doanh nghiệp cloud-first không có alert dự đoán chi phí
* Chi phí có thể vượt ngân sách 10–30% trước khi được phát hiện

*Quy trình kiểm soát thủ công:*

* Trung bình 3–5 ngày để xử lý yêu cầu phê duyệt giảm chi / cắt tài nguyên
* Tăng độ trễ trong phản ứng

*Phân tích phương sai (variance) không nhất quán:*

* Phân tích variance chủ yếu sau tháng/quý → không hỗ trợ điều chỉnh tức thời
* Khó quy trách nhiệm cho từng cost center

**Tác động định lượng ước tính:**

* Overspend trung bình: +15% so với kế hoạch năm
* Thiệt hại chi phí: ~50,000–200,000 USD/năm tùy quy mô dự án
* Thời gian nhân viên tài chính dành cho reconciliation: ~10–20 giờ/tháng

## 1.3 Stakeholders Affected and Their Concerns

**Finance Team**

* Không có công cụ theo dõi realtime → khó quản lý kế hoạch chi tiêu
* Phải dựa vào IT để giải thích phát sinh chi phí
* Lo ngại rủi ro audit và compliance

**IT/Cloud Team**

* Bị yêu cầu cắt giảm đột ngột → ảnh hưởng dịch vụ
* Không có dự báo rõ ràng để tối ưu tài nguyên
* Căng thẳng trong giao tiếp với Finance

**Management / Executive**

* Thiếu thông tin chính xác để ra quyết định
* Rủi ro hình ảnh doanh nghiệp khi overspend
* Áp lực kiểm soát chi phí trong giai đoạn tiết kiệm ngân sách

**Audit / Compliance**

* Thiếu traceability và minh bạch
* Tăng rủi ro bị phê bình hoặc trượt audit

## 1.4 Business Consequences of Inaction

Nếu không giải quyết, doanh nghiệp sẽ đối mặt với các hậu quả:

* **Chi phí vượt kiểm soát:** Không có alert kịp thời dẫn đến overspend tích lũy qua thời gian.
* **Quy trình phê duyệt rườm rà:** Chậm phản ứng với sự cố, ảnh hưởng dịch vụ.
* **Mất uy tín nội bộ:** Finance và IT không đồng thuận, đổ lỗi lẫn nhau.
* **Khó đạt mục tiêu tiết kiệm:** Không thể đáp ứng kế hoạch cắt giảm chi tiêu của ban lãnh đạo.
* **Rủi ro kiểm toán:** Thiếu dữ liệu minh bạch, khó truy vết, dễ phát hiện lỗi trong audit.

Ví dụ minh họa:

Một công ty cloud-first chi ~1 triệu USD/năm cho AWS. Nếu overspend 15%, thiệt hại tiềm năng là ~150,000 USD – đủ để thuê thêm 1–2 kỹ sư full-time.

## 1.5 Market Opportunity

Xu hướng thị trường đang hỗ trợ mạnh nhu cầu này:

**FinOps Growth:**

* Tăng trưởng hơn 20% CAGR (theo báo cáo FinOps Foundation)
* Các doanh nghiệp cloud-first coi quản lý chi phí như năng lực chiến lược

**AWS Native Tools:**

* AWS Budgets, Cost Explorer đã hỗ trợ nhưng cần tích hợp tốt hơn
* AWS phát triển Predictive Alerts để giải quyết nhu cầu này

**Competitive Advantage:**

* Doanh nghiệp áp dụng kiểm soát tự động sớm sẽ tiết kiệm chi phí, có lợi thế cạnh tranh
* Cải thiện khả năng mở rộng mà không tăng rủi ro ngân sách

# 2. Solution Architecture

## 2.1 Architecture Overview

Giải pháp đề xuất được xây dựng trên kiến trúc serverless, tận dụng các dịch vụ AWS sẵn có để giảm chi phí hạ tầng, dễ dàng mở rộng, và tăng tốc triển khai.

Hệ thống gồm bốn thành phần chính:

**1. Predictive Budget Alerting:**

* AWS Budgets với tính năng forecast-based alerting
* SNS Topic gửi thông báo khi dự báo vượt ngưỡng

**2. Automation & Controls:**

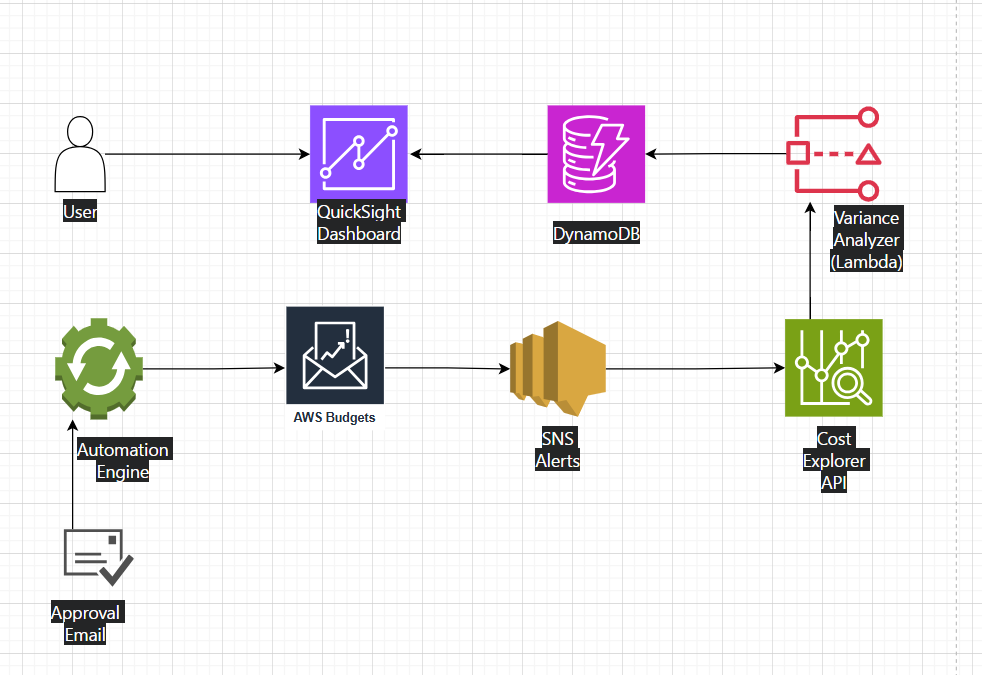
* AWS Lambda nhận sự kiện từ SNS
* Kiểm tra chính sách tự động (ví dụ block/approve)
* Gửi yêu cầu phê duyệt hoặc tự động giới hạn tài nguyên

**3. Variance Analysis Module:**

* Thu thập dữ liệu chi phí thực tế từ Cost Explorer
* So sánh với ngân sách kế hoạch (stored in DynamoDB)
* Tính toán và lưu variance

**4. Dashboard & Reporting:**

* AWS QuickSight trực quan hóa variance, alert history, dự báo chi phí
* Phân quyền truy cập theo vai trò (Finance, IT, Management)



## 2.2 AWS Services Selection and Justification

**AWS Budgets:**

* Tạo ngân sách theo cost center, project, account
* Predictive Alerts dùng machine learning forecast
* Native integration với SNS

*Justification:* Dịch vụ quản lý ngân sách tiêu chuẩn, không cần xây lại ML forecast riêng.

**AWS Cost Explorer API:**

* Truy vấn chi phí thực tế granular
* Phân tích theo service, tag, account

*Justification:* Nguồn dữ liệu chính để tính variance.

**AWS SNS / EventBridge:**

* Nhận thông báo từ AWS Budgets
* Kích hoạt Lambda Automation

*Justification:* Event-driven, đơn giản, serverless.

**AWS Lambda:**

* Xử lý sự kiện cảnh báo
* Kiểm tra policy tự động
* Thực hiện auto-cutoff hoặc escalate

*Justification:* Không cần server, tiết kiệm chi phí, dễ bảo trì.

**AWS DynamoDB:**

* Lưu dữ liệu ngân sách kế hoạch
* Store kết quả variance analysis

*Justification:* NoSQL, scale tốt, chi phí thấp, dễ tích hợp Lambda.

**AWS QuickSight:**

* Xây dashboard realtime
* User-level access control

*Justification:* Visualization native AWS, không cần self-host BI.

**IAM & KMS:**

* Quản lý truy cập và mã hóa dữ liệu

*Justification:* Đáp ứng yêu cầu bảo mật và compliance.

## 2.3 Component Interactions and Data Flow

**Step 1 – Budget Definition & Forecasting**

* Finance Team tạo ngân sách trên AWS Budgets.
* AWS Budgets tự động theo dõi chi phí và tính forecast.

**Step 2 – Alerting**

* Khi forecast vượt ngưỡng → AWS Budgets phát SNS Alert.
* SNS kích hoạt Lambda Automation.

**Step 3 – Automation & Controls**

* Lambda kiểm tra policy (ví dụ: tự động dừng non-critical resource).
* Nếu cần phê duyệt: Gửi email hoặc EventBridge để tích hợp quy trình phê duyệt.
* Log action và kết quả vào DynamoDB.

**Step 4 – Variance Analysis**

* Lambda scheduler quét Cost Explorer định kỳ (hàng ngày/tuần).
* So sánh chi phí thực tế vs kế hoạch trong DynamoDB.
* Lưu variance result.

**Step 5 – Dashboard & Reporting**

* QuickSight kết nối DynamoDB để render báo cáo.
* Hiển thị forecast, variance, alert history theo cost center.
* Finance/IT/Exec truy cập phân quyền.

## 2.4 Security Architecture and Compliance

**Principle of Least Privilege:**

* IAM Roles chuyên biệt cho Lambda, SNS, DynamoDB
* Chính sách chỉ cấp quyền cần thiết (GetCostAndUsage, Publish, InvokeFunction, Read/Write)

**Encryption:**

* KMS cho dữ liệu trong DynamoDB
* SSE (Server-Side Encryption) mặc định cho S3 (nếu dùng lưu logs)
* SSL/TLS cho API calls

**Audit Logging:**

* CloudTrail bật để log tất cả API calls
* SNS/Lambda logs gửi CloudWatch Logs
* Theo dõi tất cả hoạt động Automation & Controls

**Compliance Considerations:**

* Dữ liệu ngân sách không chứa PII → risk thấp
* Tuy nhiên vẫn tuân thủ chính sách bảo mật nội bộ
* Đáp ứng yêu cầu kiểm toán với audit logs đầy đủ

## 2.5 Scalability and Performance Considerations

**Serverless Architecture:**

* Lambda scale tự động theo số lượng alerts
* DynamoDB scale theo dung lượng data

**Cost Efficiency:**

* Pay-per-use cho Lambda và SNS
* DynamoDB On-Demand pricing để tiết kiệm

**Performance:**

* SNS → Lambda latency <1s
* Variance analysis batch → tối ưu chạy ngoài giờ hành chính
* QuickSight cập nhật dữ liệu gần realtime

**Future Scaling:**

* Dễ dàng thêm cost centers hoặc regions
* Hỗ trợ multi-account AWS Organizations

## 2.6 Integration Points with Existing Systems

**Finance Tools:**

* Export variance report CSV cho hệ thống ERP
* Dashboard QuickSight cho CFO và kế toán

**Cloud Management:**

* AWS Organizations để quản lý nhiều account
* Tagging standard để mapping chi phí

**Notification Systems:**

* Email, Slack, Microsoft Teams (SNS subscription)
* EventBridge rule cho quy trình phê duyệt tùy chỉnh

**Approval Workflow:**

* Có thể tích hợp ServiceNow, Jira cho yêu cầu phê duyệt
* EventBridge để gửi event đến hệ thống ITSM

# 3. Technical Implementation

## 3.1 Implementation Phases with Deliverables

Giải pháp được chia thành **4 pha triển khai chính**, phù hợp cho lịch trình 3–4 tuần của dự án PoC (Proof-of-Concept).

**Phase 1 – Requirements and Architecture Design**

* Thu thập yêu cầu từ Finance và IT
* Phân tích cost center, forecast needs
* Thiết kế kiến trúc tổng quan (AWS services, data flow)
* Deliverable: High-Level Architecture Document

**Phase 2 – Budget Setup and Alerting Integration**

* Tạo AWS Budgets với ngưỡng và forecast-based alerts
* Cấu hình SNS Topic để nhận thông báo
* Tích hợp EventBridge rules nếu cần routing tùy chỉnh
* Deliverable: Budget Alerting Functional

**Phase 3 – Automation & Variance Analysis Module**

* Viết AWS Lambda Functions:
  + Nhận SNS Alert → kiểm tra policy → trigger controls
  + Thu thập Cost Explorer data → tính variance → ghi DynamoDB
* Tích hợp DynamoDB schema lưu ngân sách và variance
* Deliverable: Automation Engine & Variance Analyzer ready

**Phase 4 – Dashboard & Reporting**

* Kết nối DynamoDB với AWS QuickSight
* Thiết kế dashboard với biểu đồ variance, forecast, alert history
* Phân quyền người dùng theo vai trò
* Deliverable: QuickSight Dashboard for Finance/IT

## 3.2 Technical Requirements

**Compute**

* AWS Lambda (~128–512 MB RAM)
  + Automation Engine (SNS trigger)
  + Variance Analysis (scheduled trigger)
* No EC2 instances → giảm quản trị

**Storage**

* AWS DynamoDB
  + Tables:
    - BudgetPlans
    - VarianceResults
    - AlertLogs
  + On-Demand capacity billing

**Network**

* AWS VPC Endpoints (tuỳ chọn)
  + Cho Lambda → DynamoDB secure access
* Internet access cho API AWS Cost Explorer

**IAM**

* Roles và Policies tối thiểu quyền:
  + Lambda → Cost Explorer Read
  + SNS Publish/Subscribe
  + DynamoDB Read/Write
  + QuickSight access

## 3.3 Development Approach and Methodologies

**Agile-inspired (Lightweight):**

* Chia thành 4 sprints nhỏ (1 tuần mỗi sprint)
* Mỗi sprint có planning, daily standup (nếu team >1 người), demo cuối tuần

**Infrastructure as Code (IaC):**

* Sử dụng AWS CloudFormation hoặc Terraform
* Quản lý version IaC trên Git
* Tái deploy dễ dàng (Dev, Staging, Prod)

**Code Management:**

* AWS Lambda code in Python (bổ biến và dễ tích hợp AWS SDK)
* Source control: Git
* PR review (nếu team >1)

**Best Practices:**

* Environment Variables cho config
* Logging đầy đủ (CloudWatch Logs)
* Error handling rõ ràng

## 3.4 Testing Strategy

**Unit Testing**

* Hàm Lambda phân mảnh, testable
* Mock AWS SDK calls (boto3)
* Ví dụ: kiểm tra alert parsing, kiểm tra cost comparison

**Integration Testing**

* SNS → Lambda trigger test
* Lambda → DynamoDB writes
* Cost Explorer API → Lambda

**Performance Testing**

* Kiểm tra concurrency limit Lambda (scale test)
* DynamoDB On-Demand throughput test
* Alert latency từ Budget → SNS → Lambda

**User Acceptance Testing (UAT)**

* Demo end-to-end flow cho Finance và IT
* Approval của stakeholder trước Production deploy

**Deliverables của Testing:**

* Test cases và kết quả (document)
* Logs lưu trong CloudWatch để trace

## 3.5 Deployment Plan and Rollback Procedures

**Deployment Steps**

Deploy IaC stack (CloudFormation/Terraform):

* SNS Topics
* Lambda Functions
* DynamoDB Tables  
   Configure AWS Budgets và SNS integration  
   Set IAM Roles và Policies  
   Configure QuickSight datasets và dashboard  
   Run smoke tests

**Rollback Procedures**

* IaC: rollback stack hoặc destroy/redeploy
* Lambda versioning → revert function version
* DynamoDB backup (nếu cần)
* Manual disable Budget Alerts nếu alert storm

**Deployment Environment Strategy:**

* Dev → Staging → Prod
* Tách account hoặc VPC nếu cần compliance

## 3.6 Configuration Management

**Source Control:**

* Git repo chứa:
  + IaC templates
  + Lambda code
  + Documentation

**Parameter Store/Secrets Manager:**

* Lưu API keys (nếu cần)
* Lambda environment variables

**Versioning:**

* Lambda Function Versions
* DynamoDB Schema changes versioned

**Change Management:**

* Phê duyệt pull request
* Ghi log changes vào repository
* Tag release

**Monitoring Configurations:**

* CloudWatch Alarms cho error rates
* SNS subscription health checks
* DynamoDB capacity alerts

# 4. Timeline & Milestones

## 4.1 Project Phases Breakdown

Dự án **Automated Budget Management with Predictive Alerting and Controls** được thiết kế triển khai theo **4 pha chính**, phù hợp cho thời gian 3–4 tuần thực tập (PoC).

| **Phase** | **Name** | **Duration** | **Main Activities** |
| --- | --- | --- | --- |
| 1 | Requirements & Architecture Design | 4–5 days | Thu thập yêu cầu, phân tích, thiết kế kiến trúc |
| 2 | Budget Setup & Alerting Integration | 4–5 days | Tạo AWS Budgets, cấu hình SNS/EventBridge |
| 3 | Automation & Variance Analysis | 5–6 days | Phát triển Lambda, DynamoDB, Cost Explorer, testing |
| 4 | Dashboard & UAT | 4–5 days | Thiết kế QuickSight, phân quyền, kiểm thử UAT |

Tổng thời gian ước tính: 3–4 tuần.

## 4.2 Key Milestones with Success Criteria

| **Milestone** | **Expected Date** | **Success Criteria** |
| --- | --- | --- |
| M1. Requirements Sign-off | End Week 1 | Tài liệu kiến trúc được duyệt, yêu cầu rõ ràng |
| M2. Budget & Alerting Config Completed | Mid Week 2 | AWS Budgets với forecast-based alerts hoạt động, SNS tích hợp |
| M3. Automation Module Tested | End Week 2 | Lambda tự động xử lý alerts, lưu logs DynamoDB |
| M4. Variance Analysis Module Completed | Mid Week 3 | Thu thập Cost Explorer, tính variance, lưu trữ DynamoDB |
| M5. Dashboard Ready for UAT | End Week 3 | QuickSight dashboard hoạt động, người dùng truy cập thành công |
| M6. UAT Sign-off & Handover | End Week 4 | Stakeholder đồng ý kết quả, tài liệu hướng dẫn bàn giao đầy đủ |

## 4.3 Dependencies Identification

**Internal Dependencies:**

* Finance team cung cấp kế hoạch ngân sách, cost centers
* IT team hỗ trợ access AWS Account

**AWS Services:**

* AWS Budgets cần quyền cấu hình và account có Cost Explorer enabled
* SNS / EventBridge cần tạo topic và rule trước khi Lambda deploy

**External Dependencies:**

* QuickSight license/account
* Approval từ Finance/IT trước khi chuyển pha tiếp theo

## 4.4 Critical Path Analysis

Các hoạt động trên **critical path** không được trễ:  
 **Requirements Sign-off (M1):**

* Thiếu đồng thuận yêu cầu → delay toàn dự án.  
   **Budget Setup (M2):**
* Không có alert → Lambda không test được.  
   **Automation Module (M3):**
* Delay → không thể test variance module.  
   **Dashboard UAT (M5–M6):**
* Approval muộn → dự án không bàn giao đúng hạn.

**Biện pháp quản lý:**

* Họp daily/alternate-day standups (15 phút)
* Review tiến độ mỗi cuối tuần
* Quản lý issue/tracker (Excel/Asana)

## 4.5 Resource Allocation Plan

**Project Roles:**

* Project Lead (1): Tổng điều phối, đảm bảo timeline
* Developer (1–2): Viết Lambda, IaC, testing
* Cloud Engineer (1): Setup AWS Budgets, SNS, DynamoDB
* BI/Reporting (1): Thiết kế QuickSight
* Stakeholder (Finance/IT): UAT, sign-off

**Time Allocation Example (1 Developer):**

| **Week** | **Tasks** | **Effort (hrs)** |
| --- | --- | --- |
| 1 | Requirements, design | 12–16 |
| 2 | AWS Budgets, SNS, Lambda alert handler | 16–20 |
| 3 | Variance analysis Lambda, DynamoDB schema, testing | 20–24 |
| 4 | QuickSight setup, UAT, documentation | 16–20 |

**Tools:**

* AWS Console
* CloudFormation/Terraform
* Git / CodeCommit
* Asana / Trello (task tracking)
* Slack/Email (communication)

## 4.6 Buffer Time for Risks

**Identified Risks and Buffer Mitigation:**

| **Risk** | **Probability** | **Impact** | **Mitigation** |
| --- | --- | --- | --- |
| Delay yêu cầu từ Finance/IT | High | High | Đặt buffer 2 ngày trong phase 1 |
| AWS Budgets alert config phức tạp | Medium | Medium | Buffer 1 ngày trong phase 2 |
| Lambda errors khi integrate Cost Explorer | Medium | High | Buffer 2 ngày trong phase 3 |
| Dashboard approval chậm | Medium | Medium | Buffer 1–2 ngày trong phase 4 |

**Total Buffer Recommended:** ~4–5 days.

**Practical Plan:**

* Dự kiến 20 ngày làm việc → plan cho 15–16 ngày core + 4–5 ngày buffer.
* Giảm áp lực deadline và tăng chất lượng UAT.

# 5. Budget Estimation

## 5.1 Overview

Mục tiêu phần này là ước tính chi tiết **chi phí triển khai và vận hành** cho giải pháp **Automated Budget Management with Predictive Alerting and Controls** trên AWS, phục vụ bài toán PoC 3–4 tuần và tính khả năng mở rộng sản phẩm thật.

Báo cáo sẽ chia thành:  
 Chi phí hạ tầng AWS (hàng tháng, hàng năm)  
 Chi phí phát triển một lần  
 Chi phí dịch vụ bên thứ ba (nếu có)  
 Chi phí vận hành định kỳ  
 Phân tích ROI và thời gian hòa vốn  
 Chiến lược tối ưu chi phí

## 5.2 AWS Infrastructure Costs

**Dự toán hạ tầng AWS (serverless-first):**

| **Service** | **Usage Estimate** | **Monthly Cost (USD)** | **Annual Cost (USD)** |
| --- | --- | --- | --- |
| AWS Lambda | ~100k requests/month, 128–512MB | $2–5 | $24–60 |
| AWS SNS | ~500 notifications/month | ~$1 | ~$12 |
| AWS DynamoDB | On-Demand (Low volume PoC) | $2–5 | $24–60 |
| AWS Cost Explorer API | Included in AWS billing (free tier) | $0 | $0 |
| AWS Budgets | Up to 60 budgets (~$2–3/each) | ~$10–20 | ~$120–240 |
| AWS QuickSight | Standard Edition (per reader) | $9/user/month | ~$108/user/year |
| CloudWatch Logs | Basic logs (~1–5GB) | ~$1–2 | ~$12–24 |

**Monthly total estimate (PoC):** ~$25–40 USD  
 **Annual estimate (scale to production with ~5 users):** ~$500–700 USD

**Notes:**

* AWS Budgets pricing dựa trên số budget objects.
* QuickSight pricing scale tùy số người dùng Finance/IT.

## 5.3 Development Costs (One-time)

**Dự án PoC (~3–4 tuần):**

| **Role** | **Estimated Hours** | **Rate (USD/hr)** | **Cost (USD)** |
| --- | --- | --- | --- |
| Developer / Cloud Engineer | 80–100 hrs | ~$10–15 | $800–1,500 |
| BI / Dashboard Designer | 10–20 hrs | ~$10–15 | $100–300 |

**Total Development Cost Estimate:** **~$900–1,800 USD**

**Bao gồm:**

* IaC templates (CloudFormation/Terraform)
* Lambda functions
* SNS/EventBridge integration
* DynamoDB schema
* QuickSight dashboard
* Documentation

## 5.4 Third-party Services and Licenses

AWS-native stack → không yêu cầu bên thứ ba bắt buộc.  
 **Optional:**

* Email/SMS credits (AWS SNS): ~$1–5/month nếu dùng nhiều notifications.
* BI Tools khác (PowerBI, Tableau) → *Not needed if using QuickSight*.

**Estimate:** ~$0–50 USD/year (optional).

## 5.5 Operational Costs (Ongoing)

Hạ tầng AWS (monthly): ~$25–40 USD  
 QuickSight seats (per user): ~$9/month  
 AWS Support Plan (optional):

* Basic Support: $0
* Developer Support: ~$29/month (optional)  
   Ops/Monitoring (people cost): ~1–2 hrs/month

**Ongoing Monthly Estimate:**

* Minimum: ~$25–40 (infra only)
* Typical with ~5 users + support: ~$70–100

**Annual Estimate:** ~$500–1,200 USD

## 5.6 ROI Calculation and Break-even Analysis

**Expected Benefits:**

* Avoid overspend ~10–30% annually.
* Example cloud budget: $100,000/year.
* Savings potential: $10,000–30,000/year.

**Estimated 1-year costs:**

* Setup (one-time): ~$900–1,800
* Annual infra: ~$500–700
* **Total Year-1 cost:** ~$1,400–2,500

**ROI:**

* Cost Savings (low): $10,000 – ROI ~4–7x
* Cost Savings (high): $30,000 – ROI ~12–20x

**Break-even time:**

* ~1–2 months of savings covers full PoC investment.

## 5.7 Cost Optimization Strategies

**Serverless-first Design:**

* Pay-per-use for Lambda/SNS/DynamoDB.
* No EC2 cost.
* No idle charges.

**QuickSight Licensing:**

* Start with minimum readers.
* Use free trial or Reader-only mode.

**AWS Budgets Management:**

* Consolidate budgets to fewer objects.
* Use forecast alerts instead of many static alerts.

**Monitoring/Logging Optimization:**

* Set CloudWatch log retention policy.
* Avoid excessive debug logs in prod.

**Reserved Capacity / Savings Plans:** *(if scaling)*

* Lambda savings plans for predictable usage.
* DynamoDB provisioned capacity if high volume.

**Summary of Budget Estimation**

**One-time Development Cost:** ~$900–1,800 USD  
 **Monthly Infra Cost (PoC):** ~$25–40 USD  
 **Annual Infra Cost (prod, ~5 users):** ~$500–1,200 USD  
 **Optional Third-party Licenses:** minimal to none  
 **ROI Potential:** 4–20x within 12 months  
 **Break-even:** as short as 1–2 months

Giải pháp này được thiết kế **chi phí thấp, dễ mở rộng, và hoàn vốn nhanh** – giúp doanh nghiệp chủ động quản lý ngân sách công nghệ, giảm overspend và nâng cao năng lực quản trị tài chính.

# 6. Risk Assessment

## 6.1 Overview

Phần này nhằm xác định, đánh giá và lập kế hoạch quản trị các rủi ro liên quan đến dự án **Automated Budget Management with Predictive Alerting and Controls**.

Mục tiêu:  
 Đảm bảo triển khai đúng thời gian, chất lượng, ngân sách  
 Đảm bảo vận hành lâu dài ổn định  
 Giảm thiểu các sự cố gây gián đoạn hoặc thất bại dự án

## 6.2 Risk Identification

### **6.2.1 Technical Risks**

* **R1:** AWS Budgets API/alert limitations → không đáp ứng hết logic dự kiến
* **R2:** Lambda function errors → xử lý alert không chính xác
* **R3:** Tích hợp QuickSight phức tạp → chậm delivery dashboard
* **R4:** Cost Explorer API throttling hoặc limit

### **6.2.2 Business Risks**

* **R5:** Finance team không cung cấp ngân sách kịp thời
* **R6:** Người dùng không chấp nhận quy trình tự động hóa kiểm soát
* **R7:** Thay đổi yêu cầu trong quá trình triển khai

### **6.2.3 Operational Risks**

* **R8:** Thiếu monitoring → lỗi không được phát hiện
* **R9:** Phụ thuộc IAM permissions → chậm setup
* **R10:** Thiếu tài liệu hướng dẫn → khó duy trì về sau

## 6.3 Impact Assessment and Probability Analysis

| **ID** | **Risk Description** | **Probability** | **Impact** | **Priority** |
| --- | --- | --- | --- | --- |
| R1 | AWS Budgets alert limitations | Medium | Medium | Medium |
| R2 | Lambda function errors | Medium | High | High |
| R3 | QuickSight integration complexity | Low | Medium | Low |
| R4 | Cost Explorer API throttling | Low | Medium | Low |
| R5 | Finance team delay | High | High | High |
| R6 | User acceptance issues | Medium | High | High |
| R7 | Scope changes | Medium | Medium | Medium |
| R8 | Lack of monitoring | Medium | High | High |
| R9 | IAM permission delays | Medium | Medium | Medium |
| R10 | Missing documentation | Low | High | Medium |

## 6.4 Risk Matrix with Prioritization

**Risk Matrix (Text Representation):**

| **Impact** | **Low** | **Medium** | **High** |
| --- | --- | --- | --- |
| High | - | R5, R6, R2, R8 |  |
| Medium | R3, R4, R9 | R1, R7 | R10 |
| Low | - | - | - |

**High Priority Risks:**

* R2: Lambda errors
* R5: Finance delays
* R6: User acceptance issues
* R8: Monitoring gaps

**Medium Priority Risks:**

* R1, R7, R9, R10

**Low Priority Risks:**

* R3, R4

## 6.5 Mitigation Strategies for Each Risk

| **ID** | **Mitigation Strategy** |
| --- | --- |
| R1 | Review AWS Budgets limits early, design workaround (multiple budgets, SNS filtering) |
| R2 | Unit testing, logging, CloudWatch Alarms for Lambda errors |
| R3 | Start QuickSight design early, use sample data |
| R4 | Implement API call throttling, caching results |
| R5 | Align schedule with Finance team early, set deadlines |
| R6 | User training, change management communication |
| R7 | Lock requirements baseline, controlled change request process |
| R8 | Set up CloudWatch monitoring/alarms, regular review logs |
| R9 | Pre-define IAM policies, coordinate approvals in advance |
| R10 | Write detailed handover documentation, provide training |

## 6.6 Contingency Plans

R2 – Lambda Errors:

* Rollback to last working version (Lambda versioning)
* Manual trigger option

R5 – Finance Team Delay:

* Adjust project timeline buffer (built-in 4–5 days)
* Prioritize critical budgets for pilot

R6 – User Acceptance Issues:

* Pilot phase with selected cost centers
* Gather feedback and iterate

R8 – Monitoring Gaps:

* Manual audit every week until alarms are mature
* Create on-call escalation policy

General Contingency:

* Weekly review meeting to re-prioritize tasks
* Adjust scope/phasing if critical risk materializes

## 6.7 Monitoring and Escalation Procedures

**Monitoring Approach:**

* CloudWatch Alarms for Lambda errors, SNS delivery failures
* DynamoDB capacity monitoring
* AWS Budgets alert history review

**Escalation Procedures:**

* Tier 1: Developer reviews logs, fixes minor bugs
* Tier 2: Cloud Engineer involved for IAM/infra issues
* Tier 3: Project Lead escalates to stakeholders if timeline or scope threatened

**Review Cadence:**

* Daily standup (15 min)
* Weekly risk review with project team
* Fortnightly stakeholder update

**Summary of Risk Assessment**

Đã xác định 10 rủi ro chính (technical, business, operational)  
 Đánh giá xác suất, tác động, và ưu tiên bằng ma trận rủi ro  
 Đề xuất chiến lược giảm thiểu cho từng risk  
 Lập kế hoạch contingency cho trường hợp xấu nhất  
 Xây dựng quy trình theo dõi và xử lý escalation

Giải pháp quản trị rủi ro này đảm bảo dự án PoC ~3–4 tuần đạt kết quả thành công, giảm thiểu sự cố bất ngờ, và tạo nền tảng cho mở rộng sản phẩm thật sau này.

# 7. Expected Outcomes

## 7.1 Overview

Phần này định nghĩa các **kết quả mong đợi (Expected Outcomes)** khi triển khai thành công giải pháp **Automated Budget Management with Predictive Alerting and Controls** trên AWS.

Mục tiêu:  
 Đo lường thành công rõ ràng với **success metrics**  
 Phân chia lợi ích theo **ngắn hạn, trung hạn, dài hạn**  
 Làm nổi bật cải thiện **trải nghiệm người dùng**  
 Xác định **năng lực chiến lược** mới cho tổ chức

## 7.2 Success Metrics

**Technical Success Metrics**

* 95% accuracy of forecast-based budget alerts
* Lambda execution errors <1% over 1000 invocations
* SNS delivery success rate 99%+
* DynamoDB latency <50ms average per query
* QuickSight dashboard load time <5s

**Business Success Metrics**

* ≥10% reduction in budget overspend in year 1
* ≥80% user adoption in Finance/IT teams
* ≥90% SLA compliance for alert delivery time (<1 min from threshold breach)
* ROI ≥4x within first year

## 7.3 Short-term Benefits (0–6 months)

**Improved Visibility:**

* Near real-time AWS cost monitoring
* Single dashboard for Finance/IT

**Cost Control:**

* Automatic alerts on forecast overspend
* Early warnings reduce budget surprises

**Operational Efficiency:**

* Reduce manual monthly cost checks
* Automated variance analysis

**Stakeholder Confidence:**

* Transparent, auditable budget controls
* Easy sharing of cost insights with management

## 7.4 Medium-term Benefits (6–18 months)

**Process Standardization:**

* Institutionalize automated budget alerts
* Defined escalation procedures

**Cost Savings:**

* Expected 10–30% reduction in overspend
* Better contract/Savings Plan utilization

**Integration Opportunities:**

* Link budget alerts to approval workflows
* Connect to ITSM (ServiceNow/Jira) for automated incident generation

**User Empowerment:**

* Self-service dashboards
* Finance team can adjust forecasts without IT involvement

## 7.5 Long-term Value (18+ months)

**Strategic Financial Management:**

* Predictive budget modeling
* Multi-cloud and multi-account support

**Enterprise-wide Adoption:**

* Expand controls to other departments
* Centralized cost governance

**Cost Optimization Culture:**

* Shift from reactive to proactive financial management
* Data-driven decision-making

**Sustainability and Compliance:**

* Track and manage environmental (green cloud) cost targets
* Ensure regulatory audit readiness

## 7.6 User Experience Improvements

Alerts delivered via email/SNS in near real-time  
 Intuitive QuickSight dashboards  
 Drill-down variance analysis  
 Personalized views by cost center or project  
 No-code budget setup for Finance users  
 Clear audit trail of all budget changes and alerts

## 7.7 Strategic Capabilities Gained

Predictive cost management — avoid overspend before it happens  
 Automated policy enforcement — ensure compliance without manual checks  
 Transparency — full visibility into budget allocations and usage  
 Scalability — handle multiple cost centers, accounts, or even clouds  
 Competitive advantage — faster, more informed budget decisions

**Summary of Expected Outcomes**

Định nghĩa rõ **success metrics** đo lường thành công kỹ thuật và kinh doanh.  
 Lợi ích chia theo **short-term (0–6m), medium-term (6–18m), long-term (18+m)**, giúp lập kế hoạch triển khai và mở rộng.  
 Cải thiện rõ rệt **trải nghiệm người dùng** và khả năng tự phục vụ.  
 Tạo ra **năng lực chiến lược** giúp doanh nghiệp quản lý chi phí chủ động, minh bạch và hiệu quả hơn.

**Appendices**

# Appendix A. Technical Specifications

## AWS Services and Configurations

* **AWS Budgets:**
  + Forecast-based alerts
  + SNS integration for notifications
  + Monthly budget thresholds with variance margin
* **AWS Lambda:**
  + Runtime: Python 3.10
  + Memory: 128–512MB (depends on function)
  + Triggers: SNS, EventBridge (scheduled)
  + Environment Variables for configuration
  + Logging: CloudWatch Logs
* **AWS SNS:**
  + Topics for alert routing
  + Email/HTTP subscriptions
  + Delivery retry policy
* **AWS DynamoDB:**
  + On-Demand billing
  + Tables:
    - BudgetPlans
    - VarianceResults
    - AlertLogs
  + Partition/Sort Keys defined for query performance
* **AWS Cost Explorer API:**
  + Scheduled cost retrieval
  + Granularity: Daily/Monthly
  + Filtering by cost center, service
* **AWS QuickSight:**
  + Standard Edition
  + Datasets connected to DynamoDB
  + Dashboards with variance analysis, forecast trends

## IAM Roles and Policies

* Least-privilege access principles
* Lambda Execution Role:
  + DynamoDB: Read/Write
  + SNS: Publish/Subscribe
  + Cost Explorer: ReadOnly
* QuickSight Access:
  + Reader/Author as per user roles

**Security and Compliance**

* Data encryption at rest (DynamoDB, SNS)
* IAM role separation for dev/test/prod
* CloudWatch Alarms on error rates
* Audit logging via AWS CloudTrail

**Appendix B. Cost Calculations**

**Example Monthly Estimate (PoC):**

| **Service** | **Monthly Cost (USD)** | **Notes** |
| --- | --- | --- |
| Lambda | ~$2–5 | Pay-per-invocation |
| SNS | ~$1 | Alerts ~500 msgs/month |
| DynamoDB | ~$2–5 | Low-volume on-demand billing |
| AWS Budgets | ~$10–20 | ~5–10 budget objects |
| QuickSight | ~$9/user | ~1–5 users |
| CloudWatch Logs | ~$1–2 | Basic logging needs |
| **Total (approx)** | **$25–40** | For PoC phase |

**Example Annual Estimate (Production Scale, ~5 users):**

| **Service** | **Annual Cost (USD)** |
| --- | --- |
| Lambda | ~$60 |
| SNS | ~$12 |
| DynamoDB | ~$60 |
| AWS Budgets | ~$120–240 |
| QuickSight | ~$540 (~5 users) |
| CloudWatch Logs | ~$24 |
| **Total** | **~$500–700** |

**One-time Development Cost Estimate:**

* Developer/Cloud Engineer: ~$900–1,500
* BI/QuickSight Design: ~$100–300
* **Total:** ~$900–1,800 USD
* AWS Budgets Documentation:  
  <https://docs.aws.amazon.com/cost-management/latest/userguide/budgets-managing-costs.html>
* AWS Lambda Documentation:  
  <https://docs.aws.amazon.com/lambda/latest/dg/welcome.html>

# Appendix D. References

* AWS DynamoDB Documentation:  
  <https://docs.aws.amazon.com/amazondynamodb/latest/developerguide/Introduction.html>
* AWS SNS Documentation:  
  <https://docs.aws.amazon.com/sns/latest/dg/welcome.html>
* AWS Cost Explorer API:  
  <https://docs.aws.amazon.com/aws-cost-management/latest/APIReference/>
* AWS QuickSight Documentation:  
  <https://docs.aws.amazon.com/quicksight/latest/user/welcome.html>
* AWS Pricing Calculator:  
  <https://calculator.aws.amazon.com/>