

Software Project Economic Analysis and Decision-Making Tool Requirement Document

I. Introduction

(A) Document Purpose

This document defines the functional requirements, technical specifications, and user scenarios of the "Software Project Economic Analysis and Decision-Making Tool". It provides a comprehensive technical specification for the course group project development through modeling tools such as UML diagrams, data flow diagrams, and ER diagrams.

(B) Project Background

- Educational Need:** Software engineering courses require practical tools to help students understand economic decision-making methods in each phase of the SDLC.
- Industry Pain Point:** Small and medium-sized teams lack integrated tools to support the full process of cost estimation, budget management, and risk analysis.
- Technical Trend:** Improving economic decision-making efficiency through visualization and algorithm optimization has become an industry consensus.

II. Overall Requirement Analysis

(A) System Context Diagram



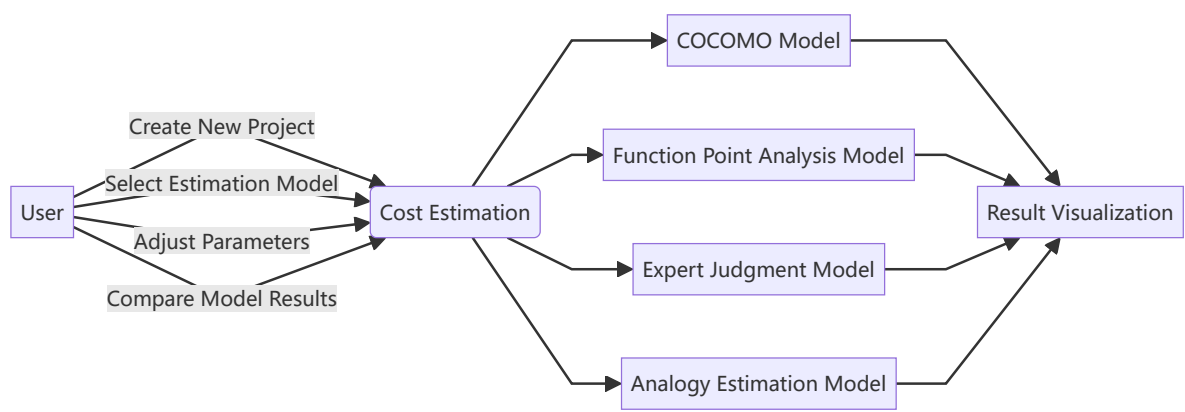
(B) System Boundaries and Environment

- Internal Entities:** Four core modules: Cost Estimation, Budget Management, Risk Management, and Resource Optimization.
- External Interfaces:**
 - Data Import Interface: Supports CSV/Excel formatted historical project data.
 - Report Export Interface: Generates PDF/PPT formatted analysis reports.
 - Teaching Evaluation Interface: Outputs student practice scores to the educational administration system.

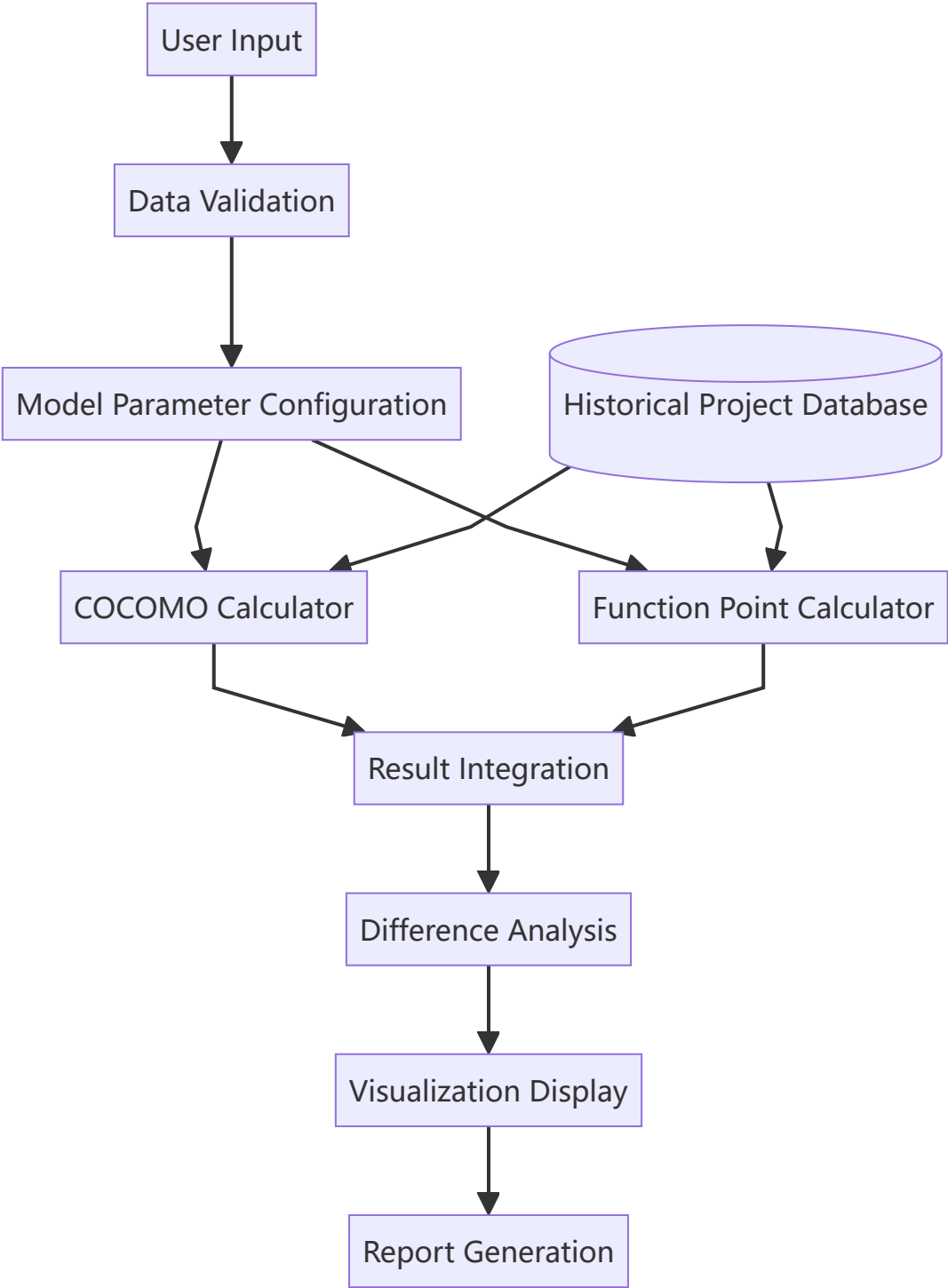
III. Detailed Functional Requirements Specifications

(A) Cost Estimation Module

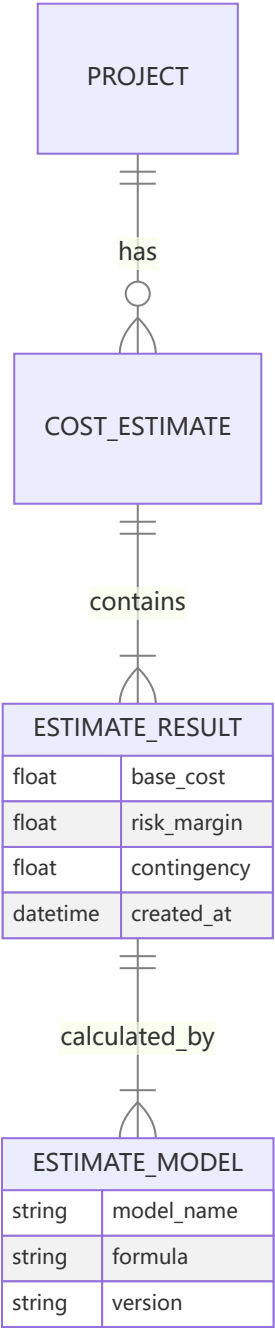
1. Use Case Diagram



2. Data Flow Diagram (DFD)

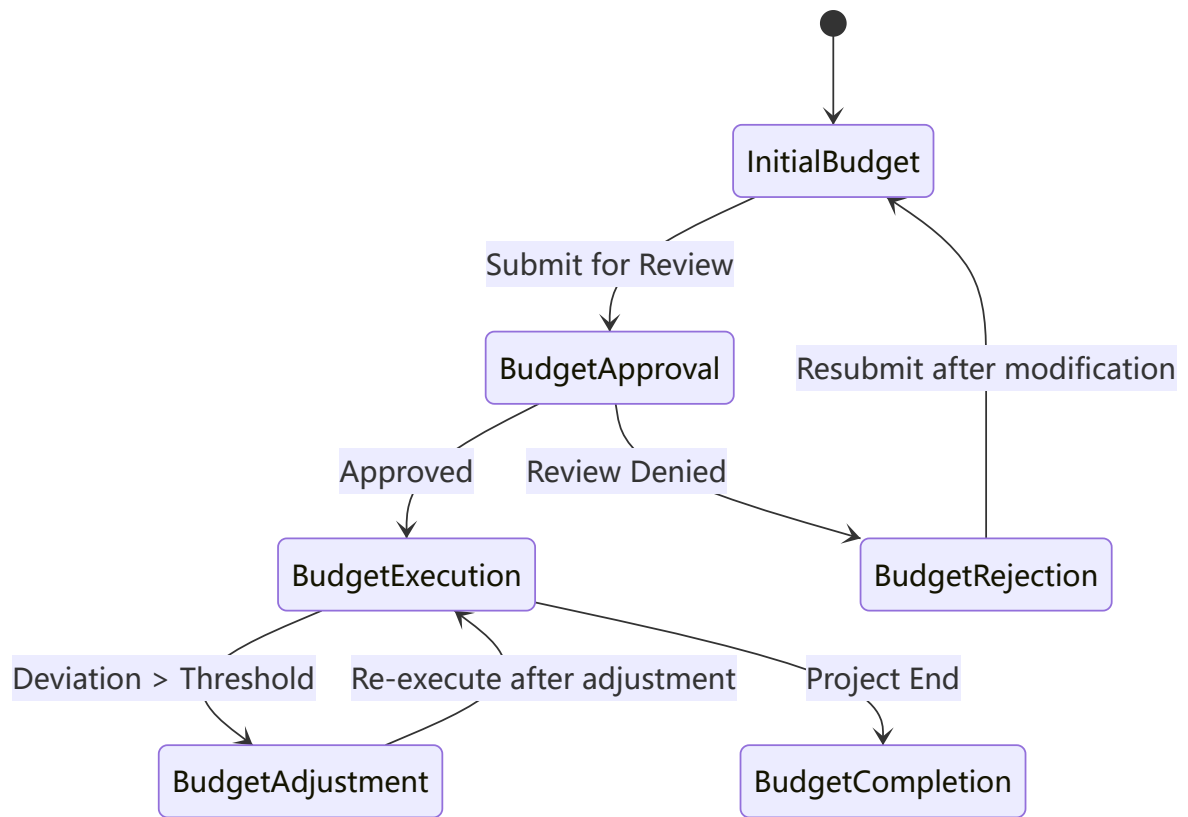


3. Data Model (ER Diagram)

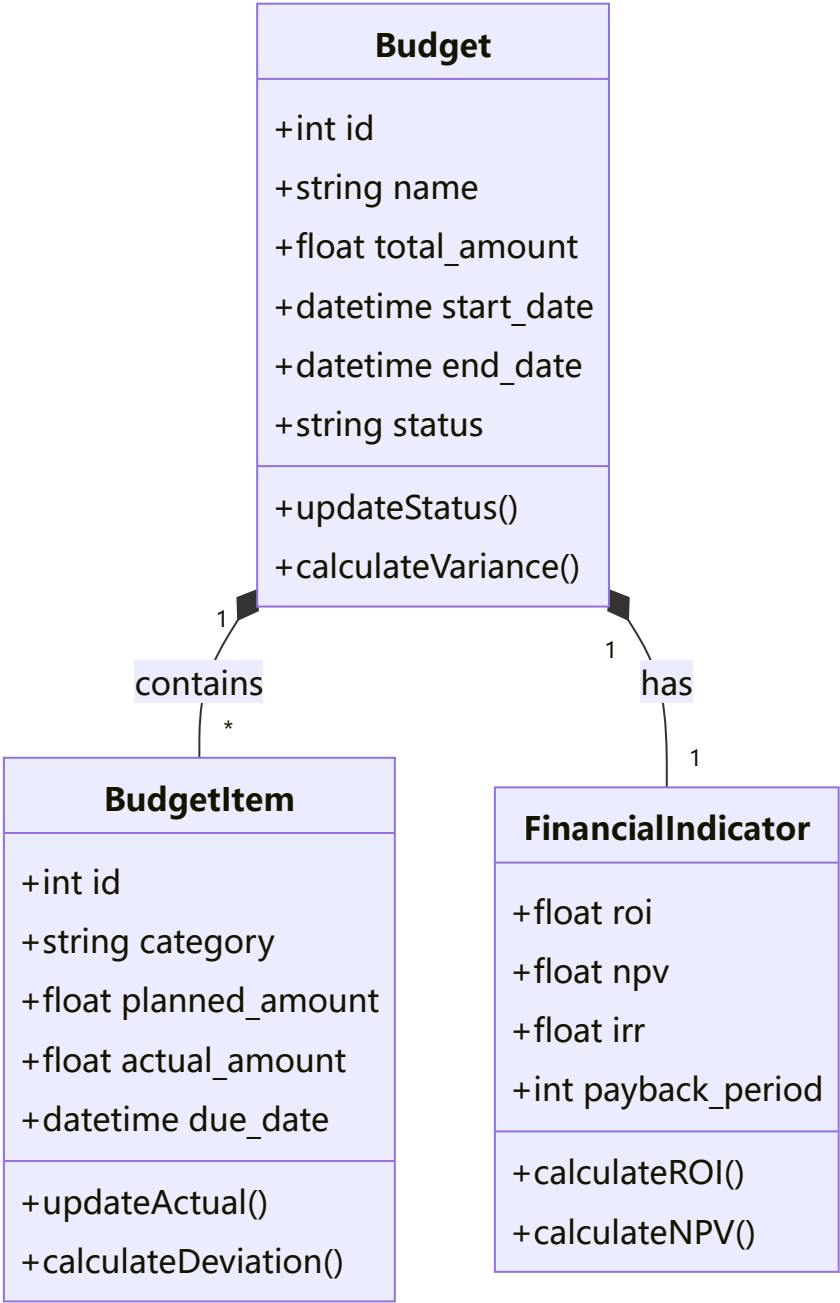


(B) Budget Management Module

1. State Diagram

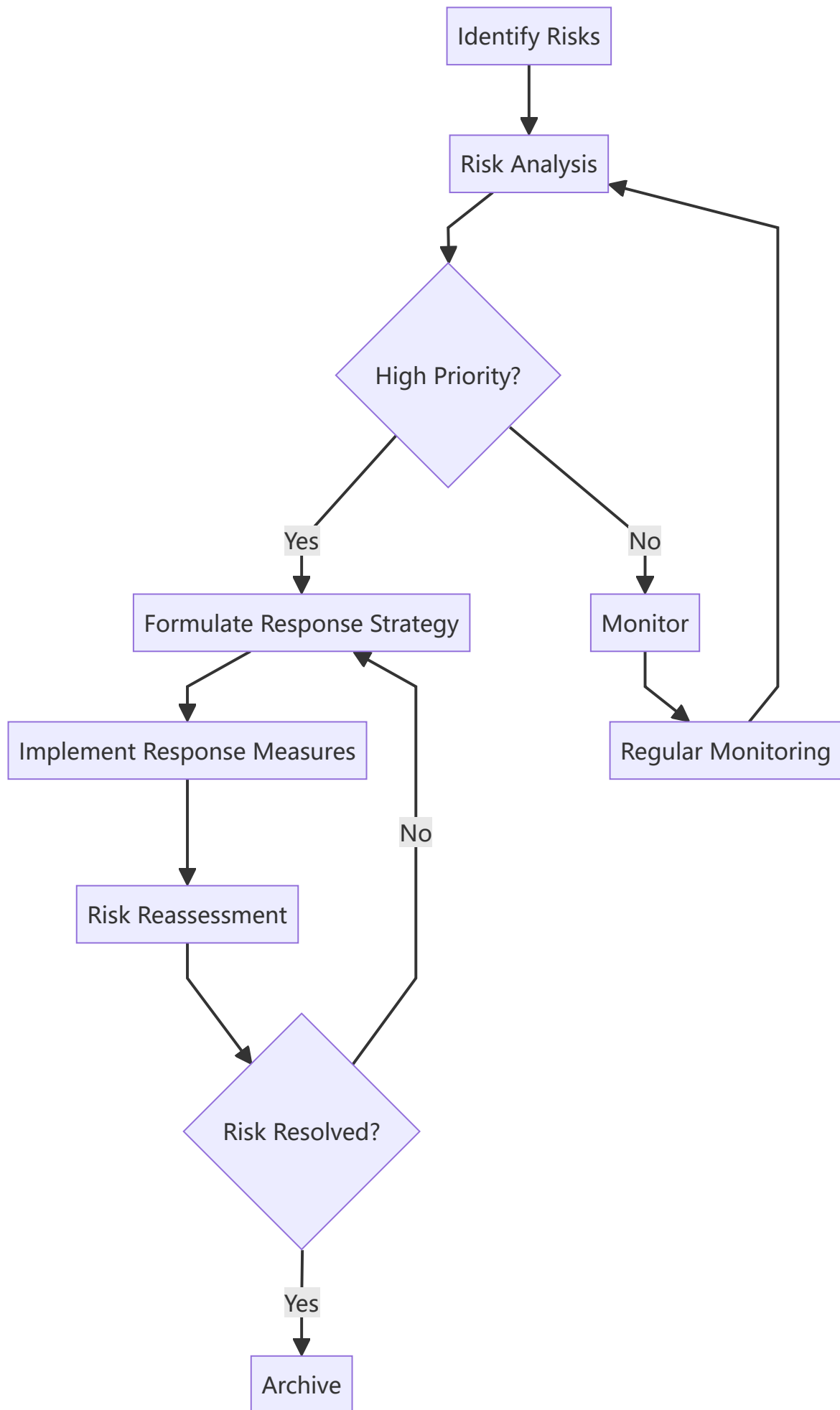


2. Class Diagram

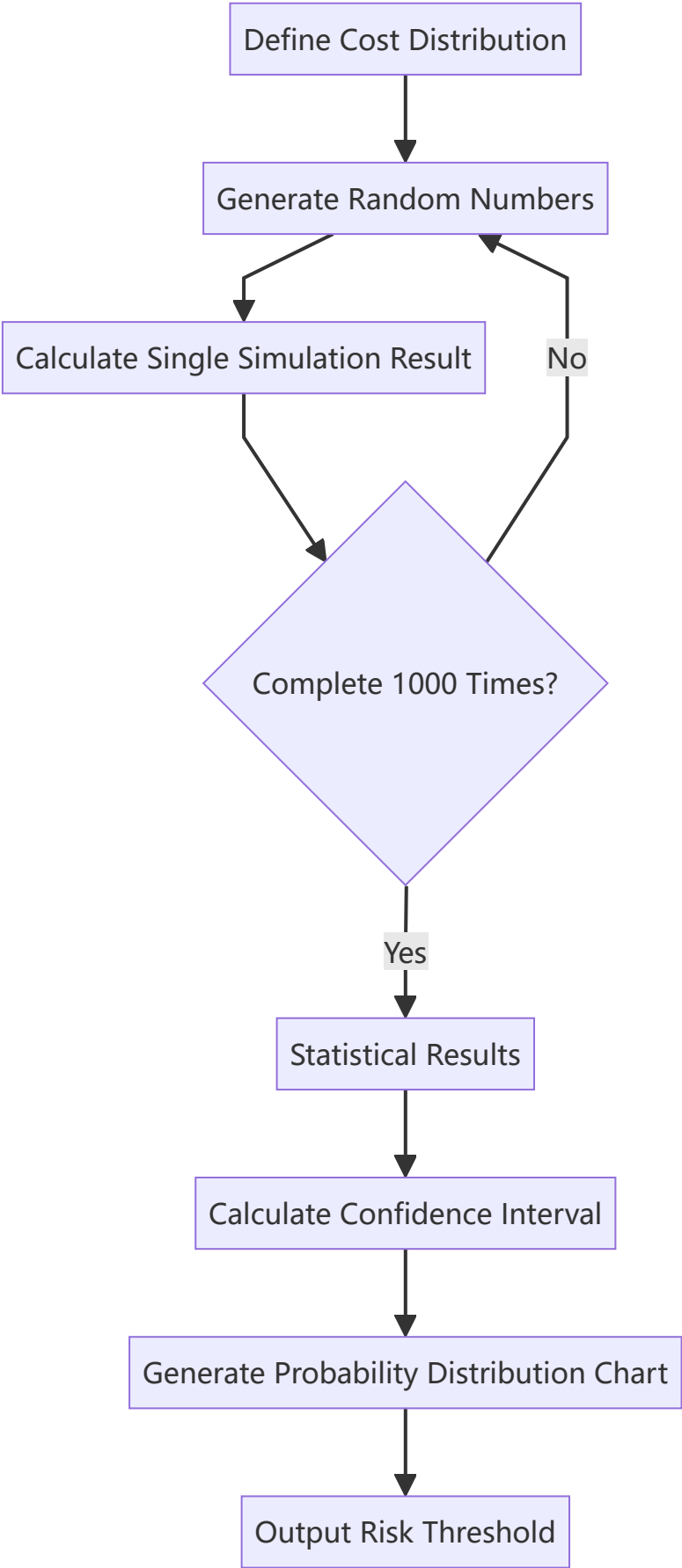


(C) Risk Management Module

1. Activity Diagram

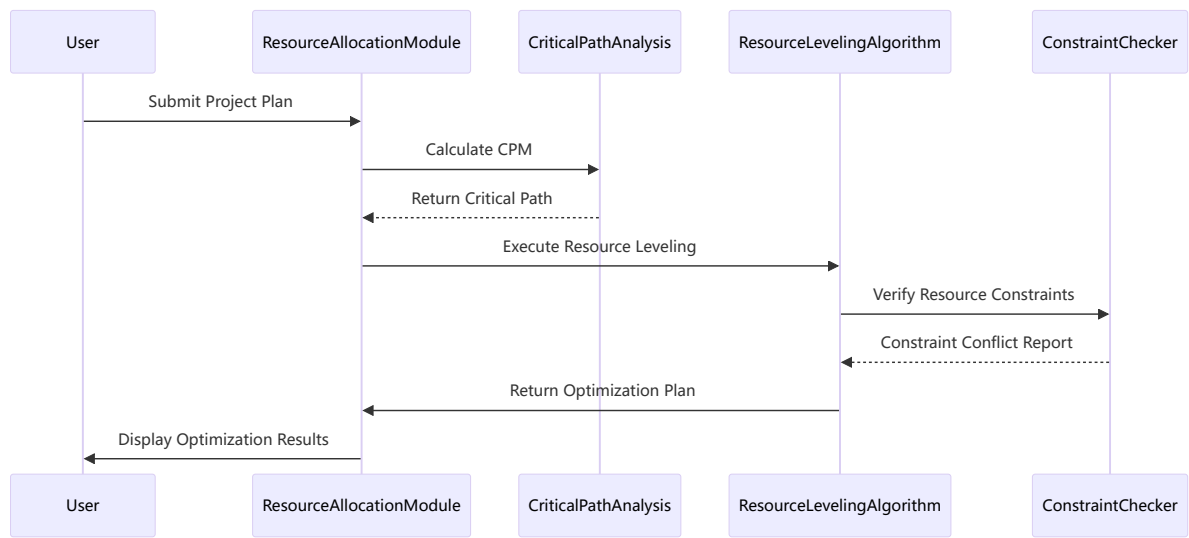


2. Algorithm Flowchart (Monte Carlo Simulation)

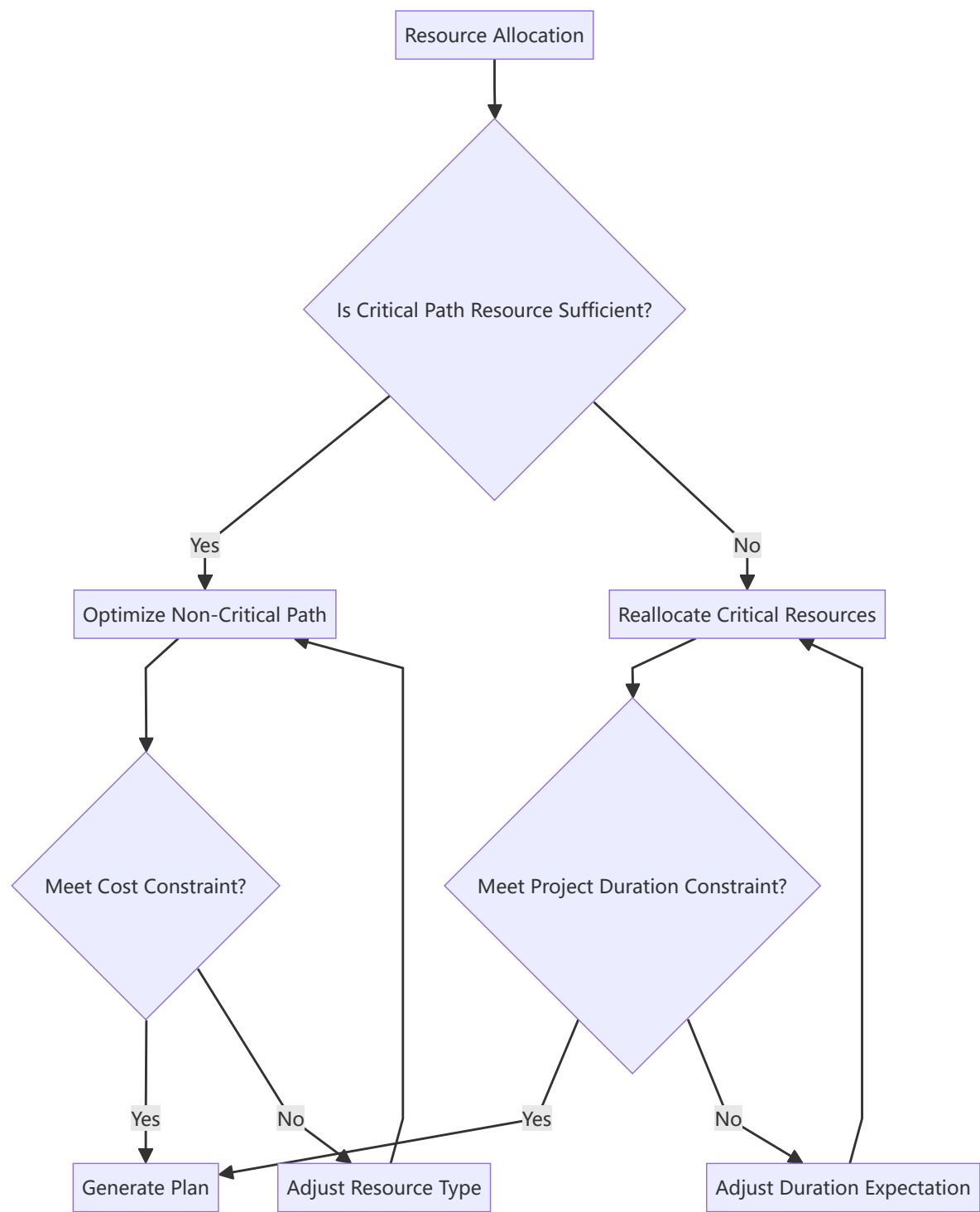


(D) Resource Optimization Module

1. Sequence Diagram



2. Decision Tree (Resource Allocation Strategy)



IV. Non-Functional Requirements Specifications

(A) Performance Requirements9

Indicator	Requirement	Testing Method
System Response Time	≤2s (normal operations)	Automated interface testing
Monte Carlo Simulation Time	≤15s	Performance testing tools
Concurrent Users	≥20 users	Load testing

Indicator	Requirement	Testing Method
Data Import Throughput	≥500 records/min	Batch data import testing

(B) Usability Requirements

1. Interface Design Standards:

- Complies with WCAG 2.1 AA accessibility standards.
- Key operation paths ≤3 steps.
- Responsive design supporting resolutions above 1024×768.

2. Help System:

- In-context help embedded in each functional module.
- Complete user manual (PDF/online).
- Tutorial videos for typical cases.

(C) Security Requirements

1. Data Security:

- Encrypted storage for sensitive data (e.g., user authentication information).
- Data backup strategy: daily full backup + real-time incremental backup.
- Access control: Role-Based Access Control (RBAC).

2. System Security:

- Prevention of SQL injection and XSS attacks.
- Regular security vulnerability scans (at least monthly).
- Abnormal login detection and alerting.

V. Data Requirements Specifications

(A) Data Dictionary (Partial)

Data Item	Type	Description	Constraints
project_id	UUID	Project unique identifier	Primary key, not null
team_size	int	Team size (number of people)	1-100
code_lines	int	Number of code lines (KLOC)	0-1000
effort	float	Workload (person-months)	>0
cost_estimate	float	Cost estimate (CNY)	>0
risk_prob	float	Risk probability (0-1)	0≤x≤1

(B) Data Flow Matrix

Source Module	Target Module	Data Items
Cost Estimation	Budget Management	Base cost, risk reserve
Budget Management	Risk Management	Budget baseline, cost threshold
Risk Management	Resource Optimization	Risk response measures, resource requirements
Resource Optimization	Budget Management	Optimized resource configuration, cost changes

VI. Interface Requirements Specifications

(A) External Interfaces

1. Data Import Interface:
- Format: CSV/Excel.
 - Field mapping: Automatic identification or manual mapping.
 - Validation rules: Data integrity check, type validation.
2. Report Export Interface:
- Formats: PDF/PPT/Excel.
 - Template configuration: Supports custom report templates.
 - Permission control: Restricts export content based on user roles.

(B) Internal Interfaces

1. Inter-module APIs
- :
- Cost Estimation → Budget Management: `/api/v1/budget/estimate`.
 - Risk Management → Resource Optimization: `/api/v1/resource/risk-adjust`.
 - Data format: JSON.
 - Error handling: Unified error code specification.

VII. Acceptance Criteria

(A) Functional Acceptance

Function Point	Acceptance Criteria	Number of Test Cases
Multi-model cost estimation	Error rate ≤10% (compared with standard cases)	15
Budget tracking	Deviation calculation accuracy 100%	10

Function Point	Acceptance Criteria	Number of Test Cases
Monte Carlo simulation	Generate probability distribution chart with 95% confidence interval	8
Resource optimization algorithm	Resource utilization improvement $\geq 15\%$	12

(B) Non-functional Acceptance

Indicator	Acceptance Criteria	Testing Tools
Response time	$\leq 2s$ (95% of requests)	JMeter
Concurrent users	20 users without 卡顿	LoadRunner
Data accuracy	Calculation error $\leq 0.1\%$	Automated test scripts
Interface usability	Average operation time $\leq 30s$	User experience testing

VIII. Appendices

(A) Abbreviation Table

Abbreviation	Full Name	Description
SDLC	Software Development Life Cycle	软件开发生命周期
COCOMO	Constructive Cost Model	构造性成本模型
EMV	Expected Monetary Value	期望货币价值
CPM	Critical Path Method	关键路径法

(B) Reference Materials

1. *Software Engineering Economics*, 4th Edition, Stephen H. Kan.
2. *Software Project Management: Processes, Methods, and Tools*, Ye Bailong.
3. IEEE Standard 1061-1998: Software Quality Metrics Methods.