

- CloudManager v1.0 - Architecture Visualization
  - Introduction
  - 1. System Overview
  - 2. Backend Architecture
  - 3. Frontend Architecture
  - 4. Data Flow
  - 5. Module Relationships
  - 6. Infrastructure & Deployment
  - 7. Authentication & Authorization Flow
  - 8. Module Dependency Matrix
  - 9. Request Lifecycle
  - 10. Database Schema Overview
  - Legend
    - Diagram Symbols
    - Color Coding
    - Architecture Layers
  - Notes

# CloudManager v1.0 - Architecture Visualization

---

## Introduction

---

This document provides comprehensive architecture visualizations for the CloudManager v1.0 Enterprise Cloud & Hosting Management Platform. The system is built with a modern tech stack featuring React 18 + TypeScript on the frontend and FastAPI + PostgreSQL on the backend.

### Tech Stack:

- **Frontend:** React 18 + Vite + TypeScript + TailwindCSS + shadcn/ui
  - **Backend:** FastAPI (async Python) + SQLAlchemy 2.0 + PostgreSQL
  - **Infrastructure:** Docker Compose + Redis + CoreDNS + Nginx + Celery
-

# 1. System Overview

High-level architecture showing all major components and their relationships.

```
Parse error on line 2:
```

```
...     subgraph client["Client Layer"]
-----^
```

```
Expecting 'SEMI', 'NEWLINE', 'SPACE', 'EOF', 'GRAPH', 'DIR',
'TAGEND', 'TAGSTART', 'UP', 'DOWN', 'subgraph', 'end', 'SQE',
'PE', '-)', 'DIAMOND_STOP', 'MINUS', '--', 'ARROW_POINT',
'ARROW_CIRCLE', 'ARROW_CROSS', 'ARROW_OPEN',
'DOTTED_ARROW_POINT', 'DOTTED_ARROW_CIRCLE',
'DOTTED_ARROW_CROSS', 'DOTTED_ARROW_OPEN', '==',
'THICK_ARROW_POINT', 'THICK_ARROW_CIRCLE', 'THICK_ARROW_CROSS',
'THICK_ARROW_OPEN', 'PIPE', 'STYLE', 'LINKSTYLE', 'CLASSDEF',
'CLASS', 'CLICK', 'DEFAULT', 'NUM', 'PCT', 'COMMA', 'ALPHA',
'COLON', 'BRKT', 'DOT', 'PUNCTUATION', 'UNICODE_TEXT', 'PLUS',
'EQUALS', 'MULT', got 'SQS'
```

# 2. Backend Architecture

Layered architecture showing the separation of concerns in the backend.

```
Parse error on line 2:
```

```
... TB     subgraph api["API Layer - Router"
-----^
```

```
Expecting 'SEMI', 'NEWLINE', 'SPACE', 'EOF', 'GRAPH', 'DIR',
'TAGEND', 'TAGSTART', 'UP', 'DOWN', 'subgraph', 'end', 'SQE',
'PE', '-)', 'DIAMOND_STOP', 'MINUS', '--', 'ARROW_POINT',
'ARROW_CIRCLE', 'ARROW_CROSS', 'ARROW_OPEN',
'DOTTED_ARROW_POINT', 'DOTTED_ARROW_CIRCLE',
'DOTTED_ARROW_CROSS', 'DOTTED_ARROW_OPEN', '==',
'THICK_ARROW_POINT', 'THICK_ARROW_CIRCLE', 'THICK_ARROW_CROSS',
'THICK_ARROW_OPEN', 'PIPE', 'STYLE', 'LINKSTYLE', 'CLASSDEF',
'CLASS', 'CLICK', 'DEFAULT', 'NUM', 'PCT', 'COMMA', 'ALPHA',
'COLON', 'BRKT', 'DOT', 'PUNCTUATION', 'UNICODE_TEXT', 'PLUS',
'EQUALS', 'MULT', got 'SQS'
```

# 3. Frontend Architecture

Modular frontend structure organized by feature modules.

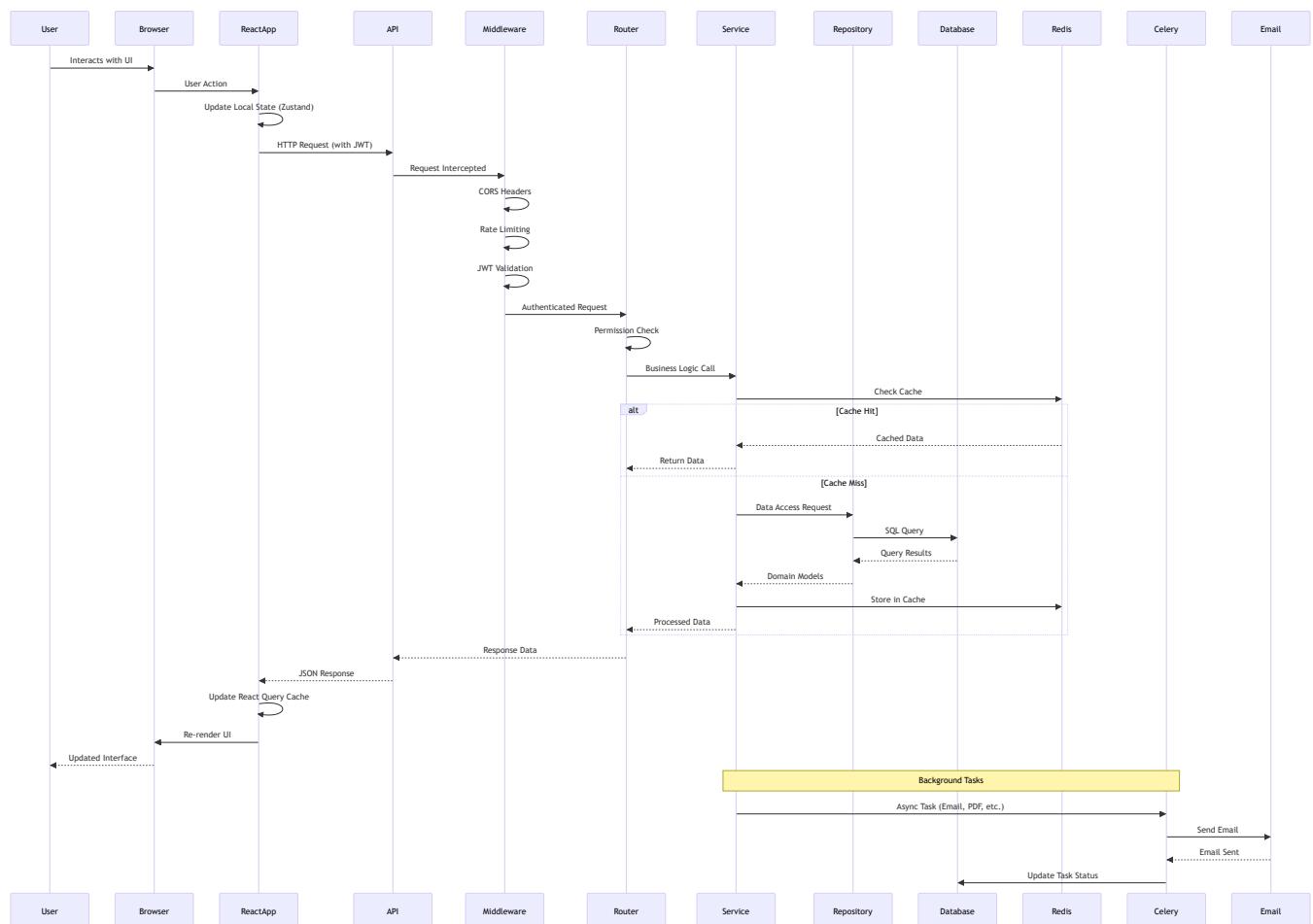
Parse error on line 2:

```
...B      subgraph entry["Entry Point"]  
-----^
```

```
Expecting 'SEMI', 'NEWLINE', 'SPACE', 'EOF', 'GRAPH', 'DIR',  
'TAGEND', 'TAGSTART', 'UP', 'DOWN', 'subgraph', 'end', 'SQE',  
'PE', '-)', 'DIAMOND_STOP', 'MINUS', '--', 'ARROW_POINT',  
'ARROW_CIRCLE', 'ARROW_CROSS', 'ARROW_OPEN',  
'DOTTED_ARROW_POINT', 'DOTTED_ARROW_CIRCLE',  
'DOTTED_ARROW_CROSS', 'DOTTED_ARROW_OPEN', '==',  
'THICK_ARROW_POINT', 'THICK_ARROW_CIRCLE', 'THICK_ARROW_CROSS',  
'THICK_ARROW_OPEN', 'PIPE', 'STYLE', 'LINKSTYLE', 'CLASSDEF',  
'CLASS', 'CLICK', 'DEFAULT', 'NUM', 'PCT', 'COMMA', 'ALPHA',  
'COLON', 'BRKT', 'DOT', 'PUNCTUATION', 'UNICODE_TEXT', 'PLUS',  
'EQUALS', 'MULT', got 'SQS'
```

## 4. Data Flow

Request/response flow from user interaction to database and back.



# 5. Module Relationships

Dependencies and interactions between different modules.

```
Parse error on line 2:
```

```
...LR      subgraph core["Core Modules"]
-----^
```

```
Expecting 'SEMI', 'NEWLINE', 'SPACE', 'EOF', 'GRAPH', 'DIR',
'TAGEND', 'TAGSTART', 'UP', 'DOWN', 'subgraph', 'end', 'SQE',
'PE', '-)', 'DIAMOND_STOP', 'MINUS', '--', 'ARROW_POINT',
'ARROW_CIRCLE', 'ARROW_CROSS', 'ARROW_OPEN',
'DOTTED_ARROW_POINT', 'DOTTED_ARROW_CIRCLE',
'DOTTED_ARROW_CROSS', 'DOTTED_ARROW_OPEN', '==',
'THICK_ARROW_POINT', 'THICK_ARROW_CIRCLE', 'THICK_ARROW_CROSS',
'THICK_ARROW_OPEN', 'PIPE', 'STYLE', 'LINKSTYLE', 'CLASSDEF',
'CLASS', 'CLICK', 'DEFAULT', 'NUM', 'PCT', 'COMMA', 'ALPHA',
'COLON', 'BRKT', 'DOT', 'PUNCTUATION', 'UNICODE_TEXT', 'PLUS',
'EQUALS', 'MULT', got 'SQS'
```

# 6. Infrastructure & Deployment

Docker Compose services and their relationships.

```
Parse error on line 2:
```

```
...      subgraph docker["Docker Compose Env"
-----^
```

```
Expecting 'SEMI', 'NEWLINE', 'SPACE', 'EOF', 'GRAPH', 'DIR',
'TAGEND', 'TAGSTART', 'UP', 'DOWN', 'subgraph', 'end', 'SQE',
'PE', '-)', 'DIAMOND_STOP', 'MINUS', '--', 'ARROW_POINT',
'ARROW_CIRCLE', 'ARROW_CROSS', 'ARROW_OPEN',
'DOTTED_ARROW_POINT', 'DOTTED_ARROW_CIRCLE',
'DOTTED_ARROW_CROSS', 'DOTTED_ARROW_OPEN', '==',
'THICK_ARROW_POINT', 'THICK_ARROW_CIRCLE', 'THICK_ARROW_CROSS',
'THICK_ARROW_OPEN', 'PIPE', 'STYLE', 'LINKSTYLE', 'CLASSDEF',
'CLASS', 'CLICK', 'DEFAULT', 'NUM', 'PCT', 'COMMA', 'ALPHA',
'COLON', 'BRKT', 'DOT', 'PUNCTUATION', 'UNICODE_TEXT', 'PLUS',
'EQUALS', 'MULT', got 'SQS'
```

# 7. Authentication & Authorization Flow

RBAC and security flow from login to resource access.

```
Parse error on line 32:  
...erify_2fa(code)    end          AuthSer  
-----^  
Expecting 'SPACE', 'NL', 'participant', 'activate', 'deactivate',  
'title', 'loop', 'opt', 'alt', 'else', 'par', 'note', 'ACTOR',  
got 'end'
```

## 8. Module Dependency Matrix

Detailed view of how modules depend on each other.

```
Parse error on line 2:  
...      subgraph layer1["Layer 1: Foundatio  
-----^  
Expecting 'SEMI', 'NEWLINE', 'SPACE', 'EOF', 'GRAPH', 'DIR',  
'TAGEND', 'TAGSTART', 'UP', 'DOWN', 'subgraph', 'end', 'SQE',  
'PE', '-'), 'DIAMOND_STOP', 'MINUS', '--', 'ARROW_POINT',  
'ARROW_CIRCLE', 'ARROW_CROSS', 'ARROW_OPEN',  
'DOTTED_ARROW_POINT', 'DOTTED_ARROW_CIRCLE',  
'DOTTED_ARROW_CROSS', 'DOTTED_ARROW_OPEN', '==',  
'THICK_ARROW_POINT', 'THICK_ARROW_CIRCLE', 'THICK_ARROW_CROSS',  
'THICK_ARROW_OPEN', 'PIPE', 'STYLE', 'LINKSTYLE', 'CLASSDEF',  
'CLASS', 'CLICK', 'DEFAULT', 'NUM', 'PCT', 'COMMA', 'ALPHA',  
'COLON', 'BRKT', 'DOT', 'PUNCTUATION', 'UNICODE_TEXT', 'PLUS',  
'EQUALS', 'MULT', got 'SQS'
```

## 9. Request Lifecycle

Complete lifecycle of a typical API request.

```
Parse error on line 1:  
flowchart TD      Sta  
^  
Expecting 'NEWLINE', 'SPACE', 'GRAPH', got 'ALPHA'
```

# 10. Database Schema Overview

High-level view of main database entities and relationships.

```
Parse error on line 1:  
erDiagram      USERS  
^  
Expecting 'NEWLINE', 'SPACE', 'GRAPH', got 'ALPHA'
```

## Legend

## Diagram Symbols

- **Rectangles**: Components, modules, or services
- **Rounded Rectangles**: Processes or actions
- **Diamonds**: Decision points or conditions
- **Circles**: Start/End points
- **Arrows**: Data flow or dependencies
- **Dashed Lines**: Optional or background processes
- **Subgraphs**: Grouped related components

## Color Coding

- **Blue**: Frontend/UI components
- **Orange**: Backend services
- **Green**: Data layer (databases, storage)
- **Purple**: Infrastructure services
- **Pink**: External services
- **Yellow**: State management

## Architecture Layers

1. **Presentation Layer**: React frontend, UI components

2. **API Layer:** FastAPI routers, middleware
  3. **Business Logic Layer:** Service classes
  4. **Data Access Layer:** Repository classes
  5. **Data Layer:** PostgreSQL, Redis
  6. **Infrastructure Layer:** Email, SMS, PDF, Storage services
- 

## Notes

---

- All modules follow a consistent layered architecture pattern
  - Maximum file size limits are enforced (150 lines for most files)
  - Each module is self-contained with its own components, hooks, services, and types
  - Infrastructure services are abstracted and can be swapped (e.g., SendGrid ↔ SMTP)
  - All database operations use async/await for better performance
  - Caching is implemented at multiple levels (Redis, React Query)
  - Security is enforced at multiple layers (JWT, RBAC, rate limiting, CORS)
- 

**Last Updated:** January 2026

**Version:** 1.0.0