AI-Driven Design Engine System Design Document

Executive Summary

Project Type: Web Application

A web-based design engine that converts structured inputs into visual representations and design artifacts.

Core Features

- Diagram-to-image rendering

- Mermaid code parsing

- visual workflow generation

- design validation

- exportable assets

- SDK integration

Constraints

- Performance

- Cost

- Security

- Compliance

- Time-to-market

User Scale: Prototype / Internal MAU, 10 concurrent users

System Overview

Functional Goals

- Render Mermaid diagrams into images

- Generate visual workflows

- Validate design artifacts

- Export design assets

Non-Functional Requirements

- High performance

- Scalability

- Security

- Compliance

Primary User Personas

- Design Engineers

- Project Managers

Architecture Design

The AI-Driven Design Engine will be built using a microservices architecture.

System Components

Component	Responsibility	Technologies	Interfaces
API Gateway	Handle incoming requests and route them to the appropriate service	NGINX, AWS Lambda	RESTful API
Mermaid Parsing Service	Parse Mermaid code and generate visual representations	Node.js, Mermaid.js	RESTful API
Visual Workflow Service	Generate visual workflows	Node.js, D3.js	RESTful API

Database Design

Database Type: Relational

High availability, high performance

Table: users

Column	Type	Nullable	Description
id	integer	False	Unique user ID
username	string	False	Username

Table: designs

Column	Type	Nullable	Description
id	integer	False	Unique design ID
user_id	integer	False	Foreign key referencing users table

Cost Estimation

Cost Item	Monthly Cost	Rationale
Infrastructure costs	$1000.0	Estimated infrastructure costs

Testing & QA Strategy

Unit testing – Individual components

Integration testing – Integrated components

End-to-end testing – Entire system

Load and stress testing – Entire system

Security testing – Entire system

Appendices

Glossary

Mermaid: A markdown-based diagramming language

References

Mermaid documentation

AWS documentation

This document is confidential and not for public distribution

System Architecture & Diagrams

System Architecture Diagram (Mermaid Code):

```
flowchart LR
User -->| Requests | API_Gateway
API_Gateway -->| Routes | Mermaid_Parsing_Service
Mermaid_Parsing_Service -->| Parses | Mermaid_Code
Mermaid_Code -->| Generates | Visual_Representation
```

Visual_Representation --> API_Gateway
API_Gateway --> User

User Flow Diagram (Mermaid Code):

```
flowchart TD
Start --> Login
Login --> Design_Flow
Design_Flow --> End
```

Database ER Diagram (Mermaid Code):

```
erDiagram
users {
int id
string username
}
designs {
int id
int user_id
}
users ||--o{ designs : has
```