Test creaeting pdf from markdown

System Design Dive Deep

A comprehensive system design learning repository with structured sessions, case studies, and hands-on materials.

Using Obsidian with This Repository

This repository is designed to work seamlessly with Obsidian, a powerful knowledge management tool.

Setup

1. **Install Obsidian**: Download from obsidian.md
2. **Open as Vault**:
    - Launch Obsidian
    - Click "Open folder as vault"
    - Navigate to: `system-design-dive-deep/SystemDesign/SystemDesign/`
    - This is your Obsidian vault root
3. **Recommended Plugins** (Optional):
    - **Mermaid** (built-in): Already supported for diagrams
    - **Dataview**: For dynamic content queries
    - **Excalidraw**: For hand-drawn diagrams
    - **Git**: For version control within Obsidian

Repository Structure

```
SystemDesign/SystemDesign/
├── Sessions/              # Topic-based learning sessions
│   ├── Cache/
│   │   ├── Cache.md              # Main concepts and diagrams
│   │   ├── Materials.md          # Curated learning resources
│   │   ├── Discussion Topics.md  # Discussion questions
│   │   └── demos/                # Code examples and POCs
│   └── [Other Topics]/
│       └── ...
```

📚 Session Topics

TOPIC	DURATION	LEVEL	KEY CONCEPTS	DISCUSSION	MATERIALS
Cache	60 min	Intermediate	Cache layers, eviction policies (LRU/LFU), write strategies, cache invalidation	Discussion Topics	Materials
Load Balancer	60 min	Intermediate	L4 vs L7 load balancing, algorithms (round-robin, least connections, IP hash), health checks	Discussion Topics	Materials
Circuit Breaker	60 min	Intermediate-Advanced	State machine (closed/open/half-open), bulkhead pattern, retry with backoff, timeouts	Discussion Topics	Materials
Consistent Hashing	60 min	Intermediate	Hash ring, virtual nodes, key redistribution, replication	Discussion Topics	Materials
Distributed Transactions	60 min	Intermediate-Advanced	2PC/3PC, Saga pattern, compensation, outbox pattern,	Discussion Topics	Materials

TOPIC	DURATION	LEVEL	KEY CONCEPTS	DISCUSSION	MATERIALS
			eventual consistency		
Message Queue	60 min	Intermediate	Kafka vs RabbitMQ vs SQS, pub/sub patterns, event-driven architecture	Discussion Topics	Materials
Rate Limiter	60 min	Intermediate	Token bucket, leaky bucket, sliding window, distributed rate limiting	Discussion Topics	Materials
Service Discovery	60 min	Intermediate	Client-side vs server-side discovery, service registry, health checking, DNS-based discovery	Discussion Topics	Materials

Navigation Tips

- **Wiki Links**: Use `**Topic Name**` to link between notes
- **Backlinks**: See which notes reference the current note in the right sidebar
- **Graph View**: Visualize connections between topics (Ctrl/Cmd + G)
- **Quick Switcher**: Jump to any note quickly (Ctrl/Cmd + O)

Working with Sessions

Each session topic follows a consistent structure:

1. **Main Document**: Core concepts with Mermaid diagrams
2. **Materials**: Curated articles, videos, and documentation (best of the best only)
3. **Discussion Topics**: Thought-provoking questions for deeper understanding
4. **Demos Folder**: Practical code examples and configurations

Mermaid Diagrams

All diagrams use Mermaid syntax and render automatically in Obsidian:

```
graph LR
    Client ⟶ LoadBalancer
    LoadBalancer ⟶ Server1
    LoadBalancer ⟶ Server2
```

Best Practices

- **Link Liberally**: Create connections between related topics using `**links**`
- **Use Tags**: Add `#systemdesign`, `#architecture`, etc. for organization
- **Daily Notes**: Keep learning logs and insights in daily notes
- **Templates**: Create templates for new session topics (see existing structure)

Contributing

When adding new topics:

1. Follow the existing folder structure
2. Use Obsidian markdown with Mermaid for diagrams
3. Curate only the best learning materials
4. Include practical discussion topics
5. Add code examples in the `demos/` folder

Resources

- Obsidian Documentation
- Mermaid Syntax
- Markdown Guide

Note: This is a public-facing repository. For internal training materials, see the `system-design-internal` repository.