<Project Name>

Solution Architecture Document (SAD)

Author	:	Ashraf Fouad Ayoub
Approver	:	
Version	:	
Last Updated Date	:	13 January, 2023

Change History

Author	Changes	Version	Date
			dd-MMM-yyyy

References

#	Author	Title / Description / Link	Version / Date
[1]			dd-MMM-yyyy

Table of Contents

Table of Diagrams

Table of Tables

1 Solution Overview

<You need to briefly introduce the solution in a couple of paragraphs, describing the functioning of the solution and its different components at a very high level. It's nice to add a high-level block diagram showing various components in one place.>

1.1 Solution Purpose

< This provides a brief about a business concern that the solution is solving and the justification to build a given solution.>

1.2 Solution Scope

< This states the business scope that the proposed solution will address. Clearly Describes out-of-scope items that the solution will not accommodate.>

1.2.1 In Scope

1.2.2 Out of Scope

1.3 Solution Assumptions

< List down all the assumptions based on which solution architect came up with the solution— for example, minimum network bandwidth availability.>

1.4 Solution Constraints

< List all technical, business, and resource constraints. Often, constraints come from industry and government compliances, which need to be listed in this section. You can also highlight the risk and mitigation plan.>

1.5 Solution Dependencies

< List all upstream and downstream dependencies. For example, an e-commerce website needs to communicate with a shipping system such as UPS or FedEx to ship a package to customers.>

1.6 Key Architecture Decision

< List major problem statements and the corresponding proposed solution options. Describe the pros and cons of each option, why a particular decision was made, and the rationale behind it.>

2 Business View

< provide a high-level overview of the business capabilities and requirements that the solution will address. This section only contains an abstract view of requirements. Detailed requirements need to be a part of a separate requirements document. However, the external link of the requirements document can be provided here.>

2.1 Business Capabilities

< Provide a brief description of the business capabilities for which the solution is being designed. Make sure to include the benefits of capabilities and how they will address customer needs.>

2.2 Key Business Requirements

< List all key business concerns that the solution is going to address. Provide a high-level view of key requirements and add a reference to the detailed requirements document.>

2.2.1 Key Business Processes

< Solution architects should show key processes with a business process document.>

2.2.2 Business Stakeholders

< List stakeholders who are directly or indirectly impacted by the project. This includes sponsors, developers, end users, vendors, and partners.>

2.3 Non-Functional Requirements

2.3.1.1 Scalability

< How can the application scale as workloads fluctuate? (For example, scale from 1,000 transactions per second to 10,000 transactions per second in a given day or month.)>

2.3.1.2 Availability and Reliability

< What is the acceptable downtime for system availability? (For example, 99.99% availability or 45 minutes' downtime per month.)>

2.3.1.3 Performance

< What is the performance requirement? Where can the system handle the load increase without impacting the end user experience? (For example, the catalog page needs to load within 3 seconds.)>

2.3.1.4 Portability

< Can the application run on multiple platforms without any additional work? (For example, the mobile app needs to run in the iOS and Android operating systems.)>

2.3.1.5 Capacity

<What is the maximum workload that the application can handle? (For example, the maximum number of users, the number of requests, the expected response time, and the expected application load.)>

2.4 Solution Constraints

2.5 Solution Dependencies

2.6 Key Architecture Decision

3 Conceptual Solution Overview

<The conceptual solution overview section provides an abstract-level diagram that captures a big-picture view of the whole solution, including business and technical aspects. It provides a basis for analyses and trade-off studies to help refine and optimize the solution architecture in sufficient detail, to support solution design and implementation.>

3.1 Conceptual and Logical Architecture

4 Solution Architecture

4.1 Information Architecture

<This section provides a user navigation flow to the application. At a high level, the solution architect needs to put in an application navigation structure.>

4.1.1 Information Components

4.2 Application Architecture

<This section targets the development team. It provides more implementation details upon which a software architect or development team can build a detailed design.>

4.2.1 Application Components

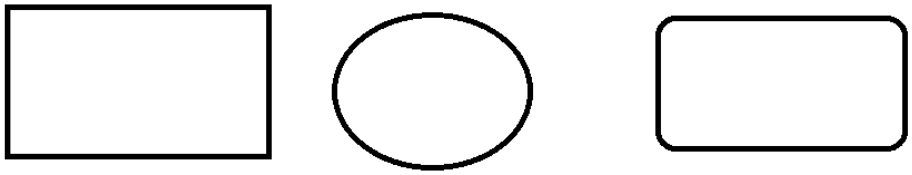

Figure 4-1: Application Components

4.3 Data Architecture

<This section is primarily utilized by the database admin and development team to understand database schemas and how tables are related. This section often includes an entity-relationship diagram (ERD) showing the relationships of entity sets stored in a database>

4.3.1 Entity Relationship Diagram (ERD)

4.3.2 Entities

Table 4-1: Table <xyx>

Field	Data Type	Allowed Values	Description

4.3.3 Data Flow and Context

4.4 Integration Architecture

< This section mainly targets vendors, partners, and other teams.>

4.4.1 Interface Component

4.5 Infrastructure Architecture

< This section is primarily targeted at the infrastructure team and system engineers. The solution architect needs to include the deployment diagram to view the logical server location and its dependencies.>

4.5.1 Infrastructure Component

4.6 Security Architecture

4.6.1 Identity and Access Management

4.6.2 Firewall configuration

4.6.3 Intrusion Prevention System (IPS)

4.6.4 Intrusion Detection System (IDS)

4.6.5 Application Security

<Example WAF, Distributed Denial of Service DDoS>

4.6.6 Data Security

<Data at rest and in transit using SSL, encryption algorithm, key management>

4.6.7 Application Threat Model

5 Solution Implementation

5.1 Development

< This section is essential for the development team. It talks about development tools, programming language, code repository, code versioning, and branching, with the rationale behind the choices.>

5.2 Deployment

< This section mainly focuses on DevOps engineers and talks about the deployment approach, deployment tools, various deployment components, and deployment checklist, with the rationale behind the choices.>

5.3 Data Migration

< This section helps the team to understand data migration and the ingestion approach, the scope of data migration, various data objects, data ingestion tools used, sources of data and data formats, and so on.>

5.4 Application Decommissioning

< This section lists existing systems that need to be decommissioned and an exit strategy for the current system if the return on investment (ROI) is not being realized. The solution architect needs to provide an approach and timeline for decommissioning the old system and carry out an overall impact assessment.>

6 Solution Management

< The solution management section is focused on production support and ongoing system maintenance across other non-product environments. The solution management section is primarily targeted at the operations management team.>

6.1 Operational Management

< system patching and upgrades of dev, test, staging, and prod environments. Tools to manage application upgrades and new releases. Tools to manage system infrastructure>

6.1.1 Monitoring and Alerting

<Operations dashboard>

6.1.2 Support and Incident Management

<Production support, SLA, incident management>

6.1.3 Disaster Recovery

<Disaster Recovery and Business Process Continuation>

6.2 User On-boarding

6.2.1 User System Requirements

7 Appendix

< can include open issues and any research data, such as the outcome of the POC, tools comparison data, and vendors' and partners' data.>

7.1 Open Items

7.2 Proof of Concept Findings