

Architecture Document Template

Use this document to explain a system architecture, the decisions behind it, and the qualities it must achieve. Organize the views so stakeholders can trace a requirement through the design and understand the remaining risks.

Complete the bracketed fields with approved information for the product, organization, or program being documented. Repeat the named forms where several tasks or subjects need separate records. Remove sections that do not apply and retain links to the material used to complete the document.

Document control	Entry
Document identifier	[Identifier used by your team]
Scope and applicable version	[Product, program, locations, and version covered]
Prepared and maintained by	[Responsible role or team]
Revision and review	[Revision, review date, and approval reference]

Introduction and Goals

Identify the system and the decisions this document supports. Keep the most important quality goals visible so readers can evaluate later design choices against them.

Field	Complete with
System purpose	[Business purpose, primary users, and architectural scope]
Requirements overview	[Major capabilities and links to the authoritative requirements]
Quality goals	[Prioritized quality goals and the reason each matters]
Stakeholders	[Stakeholder role, concern, expected contribution, and review responsibility]

Constraints

Record constraints imposed on the architecture and distinguish them from choices made by the team. Cite the source and explain the affected parts of the design.

Field	Complete with
Technical constraints	[Mandated platforms, interfaces, technologies, and compatibility limits]
Organizational constraints	[Team, delivery, operating, or sourcing constraints]
External obligations	[Applicable requirement and authoritative source or approval reference]
Design consequences	[Affected view, decision, or requirement and the resulting restriction]

Context and Scope

Describe the system boundary before decomposing its internal design. Keep business interactions and technical connections distinguishable and name external parties consistently.

Field	Complete with
Business context	[Users, neighboring systems, inputs, outputs, and boundary diagram]
Technical context	[Channels, protocols, external services, and network boundaries]
Interface catalog	[Interface identifier, counterpart, purpose, and information exchanged]
Excluded scope	[Responsibilities handled outside the system and their owner or reference]

Solution Strategy

Summarize the approaches that address the main requirements. Link to detailed views and decision records instead of repeating them in this section.

Field	Complete with
Key approach	[Architectural approach and the requirement or constraint it addresses]
System decomposition	[Main responsibilities and the rationale for their separation]
Technology choices	[Chosen technologies, alternatives considered, and decision references]
Quality approach	[How the design addresses each prioritized quality goal]

Building Block View

Repeat the building block record at each documented decomposition level. Keep the level-one view complete before refining selected components.

Field	Complete with
Block identity	[Name, responsibility, boundary, and parent block]
Interfaces	[Provided and required interfaces and their contracts]
Internal structure	[Subcomponents, relationships, and diagram reference]
Dependencies and rationale	[Upstream and downstream dependencies, design rationale, and decision links]

Runtime View

Create a scenario record for each significant interaction or failure path. Explain the order of collaboration between building blocks and show the state changes that matter.

Field	Complete with
Scenario	[Goal, trigger, participants, and starting conditions]
Interaction sequence	[Ordered messages or calls and the participating blocks]
State and outcomes	[State changes, successful outcome, and observable evidence]
Failure path	[Failure condition, propagation, recovery behavior, and related design decision]

Deployment View

Describe how software responsibilities map to infrastructure in each environment. Record topology and resource assumptions without turning this view into an installation runbook.

Field	Complete with
Environment	[Purpose, trust boundary, and applicable topology diagram]
Nodes and services	[Infrastructure node, deployed blocks, and runtime responsibility]
Communication	[Network paths, protocols, security boundaries, and dependencies]
Capacity and resilience	[Sizing basis, failure domains, redundancy, and deployment constraints]

Crosscutting Concepts

Repeat this form for concepts shared by several building blocks. State the convention once and identify where the concept is implemented.

Field	Complete with
Concept	[Security, persistence, observability, domain model, or another shared concern]
Design convention	[Rules, terminology, and expected behavior across components]
Implementation	[Affected components and supporting diagrams or references]
Exceptions	[Approved deviation, reason, consequence, and decision record]

Architecture Decisions

Repeat the decision record for each significant choice. Keep the decision status and its consequences visible so future changes can be assessed in context.

Field	Complete with
Decision identifier and status	[Identifier, proposed or accepted status, and review reference]
Context and drivers	[Problem, requirements, constraints, and decision criteria]
Alternatives and decision	[Options, comparison, chosen option, and rationale]
Consequences	[Benefits, tradeoffs, follow-up work, and affected architectural views]

Quality Requirements

Make quality requirements assessable through concrete scenarios. Link each scenario to the architectural mechanisms intended to satisfy it.

Field	Complete with
Quality scenario	[Source, stimulus, environment, affected artifact, and expected response]
Response measure	[Observable measure, target with units, and measurement method]
Design support	[Relevant building block, concept, or decision]
Verification evidence	[Test or evaluation reference, observed result, and unresolved gap]

Risks and Technical Debt

Keep a record for each known risk or design compromise. Describe the consequence and the decision needed rather than listing a vague concern.

Field	Complete with
Risk or debt item	[Identifier, affected component, and description]
Cause and consequence	[Why it exists and what may happen if it remains]
Treatment	[Mitigation, acceptance, replacement, or investigation and responsible role]
Review	[Trigger or review point, current status, and decision reference]

Glossary

Use this glossary for business and technical terms that have a specific meaning in the architecture. Keep names consistent across diagrams and the text.

Field	Complete with
Term	[Term or abbreviation used in the document]
Definition	[Meaning in this system and the scope of the definition]
Related terms	[Synonyms, distinctions, or terms that should not be confused]
Source and usage	[Authoritative reference and the views in which the term appears]