Archie

An SDLC Knowledge & Documentation Assistant

Requirements & Functional Specification

Product Definition Review – David Wright St., June 2026

This document describes the purpose, scope, and functional capabilities of Archie, a proposed tool to help teams manage and interact with documentation across the software development lifecycle (SDLC). It is intended to support a Product Definition discussion with prospective users and project teams, ahead of detailed technical design. This draft intentionally avoids naming specific underlying platforms or vendors — it focuses on the capabilities Archie needs, so the concept can be evaluated on its merits before implementation choices are locked in.

Copyright © Openthinka LLC, 2026

1. Purpose & Background

Software delivery projects generate a large and growing volume of documentation — requirements, specifications, architecture diagrams, test plans, scripts, and sample data — spread across many stages of the SDLC. Alongside project-specific documents, teams must also work against organizational and industry standards (coding standards, architectural patterns, compliance references) that live in separate repositories such as internal wikis, SharePoint sites, and reference libraries.

Today, keeping project documentation aligned with current standards, and making sense of a large and growing document set, is a largely manual effort. Archie is proposed as a tool to reduce that burden by combining project documentation with relevant standards in a structured, queryable workspace, and by using AI to help teams understand relationships between the two, check for conformance, and generate new documentation from existing material.

2. Problem Statement

- Project documentation is scattered across many formats and stages, making it hard to get a unified view of a project's SDLC artifacts.
- Standards and reference material live outside the project and change independently — teams often don't know when a standard has changed or how it affects their documents.
- Producing new SDLC artifacts (for example, a test suite derived from requirements) is manual and time-consuming, even when the source material already contains everything needed.
- Assessing whether a project's documentation conforms to a given standard is a manual review exercise today, done inconsistently across teams and projects.

3. Vision

Archie gives every project a dedicated, AI-assisted workspace that brings together project documentation and the standards that apply to it. Within that workspace, users can ask natural-language questions, check conformance to standards, and generate new documents from existing ones — with changes to standards surfaced automatically so documentation doesn't silently go stale.

4. Scope

4.1 In Scope (Target State)

- A dedicated workspace per project, containing that project's SDLC documents.
- Ingestion of standards and reference material from external repositories (web/wiki, SharePoint, and similar sources) into a workspace.
- Automated detection of changes to source standards/reference material, with notification to affected project workspaces.
- Natural-language query of a workspace's contents, including questions that relate project documents to standards.
- AI-assisted generation of new project documents from existing material (e.g., generating a user acceptance test suite from a requirements document).
- Support for documentation across all SDLC stages: requirements, specifications, architecture/design, scripts, diagrams, sample data, and test artifacts.

4.2 Out of Scope (For Now)

- Automated editing or approval of project documents without user review.
- Enforcement actions (e.g., blocking a release) based on conformance findings — Archie surfaces information, it does not gate process, at least initially.
- Real-time collaborative editing of documents within the workspace.

Copyright © Openthinka LLC, 2026

4.3 Staged Delivery

Archie is intended to be developed in stages rather than delivered all at once, so that value and technical feasibility can be validated early. A representative staging approach:

1. Stage 1 — Manual validation: Stand up a single project workspace by hand, loading one project's documents and one standards document, and validate that natural-language query and document generation produce useful results.
2. Stage 2 — Semi-automated ingestion: Introduce structured, repeatable ingestion of standards/reference sources into a workspace (still user-triggered).
3. Stage 3 — Automated sync & change notification: Introduce the automated "watcher" capability that detects changes to standards and notifies affected workspaces.
4. Stage 4 — General availability: Support for multiple concurrent project workspaces, broader user access, and refinement based on feedback from earlier stages.

The remainder of this document describes the target-state functional capabilities; not all capabilities are expected to exist in Stage 1.

5. User Roles

ID	Description	Notes
Project User	A member of a project team who works within a project's workspace — asking questions, requesting generated documents, and reviewing conformance findings.	Primary day-to-day user.
Standards Owner	Maintains a standards or reference document elsewhere (wiki, SharePoint, etc.) that one or more project workspaces depend on.	May not directly use Archie; their edits are a trigger for sync/notification.
Workspace Administrator	Sets up a project workspace, decides which standards/reference sources apply to it, and manages access.	Role may be combined with Project User on smaller teams.

6. Functional Requirements

6.1 Workspace Management

ID	Description	Notes
FR-1	The system shall allow creation of a dedicated workspace for a given project.	One workspace per project, per the proposed model.
FR-2	The system shall allow project documents to be added to a workspace by users.	Covers all SDLC stages: requirements, specs, scripts, diagrams, sample data, test artifacts.
FR-3	The system shall allow one or more standards/reference documents to be associated with a workspace.	Sourced from web/wiki, SharePoint, or similar external repositories.
FR-4	The system shall support managing access to a workspace by user or group.	Detailed access model to be defined during design.

6.2 Standards & Reference Ingestion

Copyright © Openthinka LLC, 2026

ID	Requirement	Notes
FR-5	The system shall be able to ingest documents from external sources including web/wiki pages, SharePoint, and standards reference repositories.	Ingestion method (manual upload vs. connector) may vary by stage.
FR-6	The system shall detect when a previously ingested standard or reference document has changed.	Automated detection is a later-stage capability (see §4.3).
FR-7	The system shall notify users of an affected workspace when an associated standard or reference has changed.	Notification does not imply automatic action on project documents.
FR-8	The system shall re-sync an updated standard or reference document into all workspaces that reference it.	Keeps workspace content current without manual re-upload.

6.3 Natural-Language Interaction

ID	Requirement	Notes
FR-9	The system shall allow users to ask natural-language questions about the contents of a workspace.	Includes questions spanning both project documents and standards.
FR-10	The system shall answer questions about how project documents relate to associated standards.	Example: "Does the solution architecture conform to API-led standards?"
FR-11	The system shall generate new project documents from existing documents in the workspace, based on a natural-language request.	Example: "Create a user acceptance test suite from the requirements document."
FR-12	The system should provide traceability or citations from generated answers/documents back to their source material.	Important for trust and for review by the requesting user.

6.4 Conformance & Reporting

ID	Requirement	Notes
FR-13	The system shall support checking a project document's conformance to a specified standard and reporting findings.	Findings are advisory, not enforced (see §4.2).
FR-14	The system should allow a user to review and export conformance findings for a document or workspace.	Format and destination TBD during design.

7. Example Use Cases

The following examples, drawn from the initial concept, illustrate the intended interaction style:

- "Please create a user acceptance test suite from the requirements document."

Copyright © Openthinka LLC, 2026

- "Does the solution architecture conform to API-led standards?" (assuming an API-led standards document is available among the workspace's standards.)
- "What changed in the coding standard this month, and does it affect any documents in this project?"

8. Non-Functional Requirements

ID	Requirement	Notes
NFR-1	Access to a project workspace and its contents shall be restricted to authorized users.	Aligns with typical enterprise document-security expectations.
NFR-2	The system shall not use project or standards content to train external or shared AI models.	A common enterprise requirement for AI-assisted document tools.
NFR-3	The system should scale to support many concurrent project workspaces as adoption grows beyond initial stages.	Not a Stage 1 requirement, but should not be precluded by early design choices.
NFR-4	The system should preserve an audit trail of document changes and generated outputs.	Supports governance and review, particularly for conformance findings.

9. Open Questions

The following are open as of this draft and are expected to be resolved before or during detailed design:

- Which underlying platform(s) will provide workspace, ingestion, and AI-assistant capabilities — a single platform, or a combination of complementary services?
- What mechanism will detect changes in external standards/reference sources (native platform capability vs. custom-built watcher)?
- What is the access/permission model across workspaces, standards owners, and project teams?
- What volume of documents and workspaces should the design plan for at general availability?
- What audit, retention, and data-residency requirements apply, given the standards and project content involved?

10. Next Steps

1. Circulate this document for Product Definition review with prospective users and project teams.
2. Validate Stage 1 (manual workspace test) against one real or representative project and standard.
3. Evaluate candidate underlying platforms against the functional requirements in §6.
4. Refine this document into a design-ready specification based on Stage 1 findings and platform evaluation.

Copyright © Openthinka LLC, 2026