

# Architecture Review

## Context

We need a reliable path to transform Markdown documentation into stakeholder-ready PDFs without forcing teams to copy content into slides or documents.

## Proposed Flow

1. Authors write in Markdown
2. NEXO validates and renders the document
3. Playwright generates the PDF output

## Decision

Use a Next.js application with a dedicated conversion route and Playwright-based PDF rendering.

## Component Overview

Layer	Responsibility
Web UI	Upload files, preview output, trigger generation
API route	Validate payload, apply limits, invoke rendering
PDF service	Convert Markdown HTML into paginated PDF

## Trade-offs

- Faster iteration with one codebase for UI and API
- Consistent output because rendering stays browser-based
- Higher runtime cost than a plain Markdown export

## Risks

- Large files increase render time
- Browser availability must be managed in CI and deploys

## Recommendation

Approve the architecture for the v0.1 release and focus next on templates, CLI automation, and richer Markdown fidelity.