

Identify Stale Documentation

September 10, 2026

Identify Stale Documentation

This page explains how to find documentation that may no longer reflect your project's analyzed sources. Reviewing documentation freshness helps you prioritize updates after source changes and keep user-facing guidance reliable.

Overview

Documentation can become stale when the project source changes but the corresponding explanations, guides, or reference material have not been reviewed. For example, a changed frontend component, backend service, database entity, API schema, configuration setting, or infrastructure definition may affect documentation that describes how the project works.

Atloria organizes project information around individual projects. From the **Projects** area, you can open a focused project workspace and review its documentation alongside project changes and version history. Use this workflow when a new analysis has been completed, when project changes are introduced, or before you export documentation for others to use.

Freshness review is a comparison task: you identify what changed in the analyzed project sources, locate the documentation covering that area, and decide whether the documentation still matches. Documentation is not automatically “correct” simply because it exists; you should confirm that its terminology, described behavior, and scope remain aligned with the current project.



Screenshot pending: A project workspace showing where to review documentation together with project changes and versions.

Prerequisites

Before you begin, make sure you have:

- Access to the Atloria **Projects** area.
- A project that has already been added and analyzed.
- Permission to open the project's documentation, change history, and versions.
- An understanding of the source change or release you want to review.
- Access to the project's existing documentation so you can compare it with the analyzed project information.

Step-by-Step Instructions

1. Open the project list.

Select **Projects** from the application navigation. If you are already working inside a project, select ← **Back to Projects** or ← **Back to Projects** to return to the list.

The project list is your starting point for finding the project whose documentation you need to assess. If your organization has many projects, use **Search projects...** to narrow the list. You can also use **Search across all projects...** when you need to locate a project or related information more broadly.

2. Open the project you want to review.

Select the relevant project to enter its focused project workspace. This workspace brings together the project's analyzed information, documentation, changes, and version history.

If you have not yet added the project, select **New Project** and then **Create project** before continuing. A project must be available in Atloria before you can review documentation against its analyzed sources.

 *Screenshot pending: The Projects list with the project search field and a selected project opening its focused workspace.*

3. Review recent project changes.

In the project workspace, open the project's changes area. Review the available changes to understand which part of the project was affected and whether the change is likely to alter documentation.

Pay particular attention to changes involving these analyzed categories:

- **FRONTEND** changes may affect screens, user flows, visible labels, or component behavior.

- **BACKEND** changes may affect workflows, business rules, processing behavior, or integrations.
- **DATABASE** changes may affect stored data, fields, entity descriptions, or reporting.
- **API SCHEMA** changes may affect API reference material and integration guidance.
- **CONFIG** changes may affect setup, environment, feature availability, or operational instructions.
- **DOCUMENTATION** changes may indicate that documentation has already been updated or requires review.
- **INFRASTRUCTURE** changes may affect deployment, hosting, or operational documentation.

A change is a freshness signal, not automatically proof that a document is outdated. Use it to determine which documentation topics need inspection.

4. Compare versions when you need a broader view.

Open the project's versions area and select the versions you need to compare. Comparing versions is especially useful after a release, a large analysis update, or a period with multiple changes.

Start with the newest version and compare it with the earlier version that the documentation was based on. Look for newly introduced areas, removed behavior, renamed concepts, or changed relationships between project entities.

For example, documentation may require review when a component now renders another component differently, or when a component begins using a hook or another dependency that changes the behavior described in the documentation.

5. Open the project documentation.

From the focused project workspace, open the documentation area. Use **User Guides** when you need to review user-facing guidance. If your project includes generated or technical documentation, review the relevant material for the changed project area as well.

Read the document with the change information in mind. Check whether the documentation still accurately describes:

- What the feature, screen, process, or integration does.
- The names of user-visible actions and terms.

- The sequence of steps users must perform.
- The data, configuration, or dependencies involved.
- Any limitations, prerequisites, or outcomes described.

6. Mark the documentation that needs follow-up.

Record each document that no longer matches the analyzed source information. Include the document title, affected topic, related project change or version, and the reason it needs review.

A useful review note is specific. For example: “Update the user guide because the frontend workflow changed,” rather than “Documentation may be stale.” Specific notes help the document owner understand what to verify.

7. Recheck documentation before sharing it.

After updates are made, revisit the related project change or version comparison and confirm that the revised document covers the current behavior. When the documentation is ready to share, use **Export Documentation** if you need a copy outside Atloria.



Screenshot pending: A project's changes or version comparison shown alongside the User Guides documentation area.

Tips and Best Practices

- Review documentation after every meaningful analysis update. Small changes can still affect field names, navigation steps, setup instructions, or expected outcomes.
- Start with the highest-impact categories. Changes in **API SCHEMA**, **DATABASE**, and **CONFIG** often affect multiple documents, while **FRONTEND** changes commonly affect user instructions and screenshots.
- Treat renamed concepts as potential documentation issues. A document can be misleading even when the underlying feature still works if it uses old names or describes an outdated relationship.
- Use version comparison for release reviews. It provides a more complete freshness check than reviewing one isolated change.
- Review **User Guides** from the reader's perspective. Confirm that the documented steps still match what users see and do in the application.
- Do not export documentation until you have completed the freshness review. **Export Documentation** distributes the current content; it does not by itself verify

that the content matches the latest analyzed project sources.

Related Pages

- [Manage Projects](#)
- [Review Project Changes](#)
- [Compare Project Versions](#)
- [Use User Guides](#)
- [Export Documentation](#)

Troubleshooting

I cannot find the project I need to review

Return to **Projects** and use **Search projects....** If you need to search more broadly, use **Search across all projects....** If the project is not available, confirm that it has been created and analyzed before attempting a documentation freshness review.

I can see documentation, but I do not know what may be stale

Open the project's changes area or compare versions first. Use the affected category—such as **FRONTEND**, **BACKEND**, **DATABASE**, or **API SCHEMA**—to identify the documentation most likely to require review.

The documentation looks correct, but I am unsure whether it reflects the latest analysis

Check the project's latest version and compare it with the version or change set you used for the documentation review. If newer changes exist, repeat the comparison before sharing or using **Export Documentation**.