

Understand Code Analysis

September 10, 2026

Understand Code Analysis

Code analysis examines your source material and organizes it into information that Atloria can use for AI Generation and technical documentation. Understanding this process helps you provide complete inputs and recognize what Atloria is processing before you generate documentation.

Overview

Atloria analyzes source material to identify the major parts of an application and the relationships between them. During analysis, the system can classify material across areas such as **FRONTEND**, **BACKEND**, **DATABASE**, **API SCHEMA**, **CONFIG**, **DOCUMENTATION**, and **INFRASTRUCTURE**.

Within those areas, Atloria identifies entities that are useful in technical documentation. For example, it can distinguish entities such as **COMPONENT**, **HOOK**, **PAGE**, **LAYOUT**, **ROUTE**, **CONTEXT**, **PROVIDER**, **STORE**, **CONTROLLER**, **SERVICE**, **REPOSITORY**, and **MIDDLEWARE**. This organization gives AI Generation the context it needs to describe how parts of your application work together.

You typically use code analysis before selecting **Generate Documentation** → from the **Technical Docs** area. The results help the generated documentation explain application structure, dependencies, and relationships—for example, when one component renders another component or when a component uses a hook.



Screenshot pending: The Technical Docs area showing documentation actions and the source-analysis workflow.

Prerequisites

Before you begin, make sure you have the following:

- Access to the relevant project in Atloria.

- Source material available for the project.
- Permission to work in the **Technical Docs** area.
- A clear understanding of whether you want to generate new documentation or work from existing documentation.
- Enough time for the generation process to complete without closing or deleting the work in progress.

Step-by-Step Instructions

1. Open your project in Atloria.
2. Select **Technical Docs** to open the technical documentation workspace.
3. Review the available starting options in the workspace.
4. Select **Add code examples** if you need to provide source examples that will help describe implementation details.
5. Select **Or import existing documentation** if you want Atloria to work with documentation you already have.
6. Select **View Source** when you need to inspect the source material associated with the documentation.
7. Select **Source** in the editor toolbar to display the source-focused view.
8. Select **Technical** in the editor toolbar to return to the technical documentation-focused view.
9. Use the language selector in the source editor if you need the displayed source content to use the appropriate language formatting.
10. Review the source content for completeness before starting generation.
11. Check that the source includes the parts of the application you expect Atloria to analyze, such as frontend elements, backend services, database-related material, configuration, or API schemas.
12. Select **Generate Documentation** → when the source material is ready for analysis and documentation generation.
13. Watch for the status **4. Generation Running** to confirm that Atloria is processing the submitted material.

14. Wait for generation to finish before making decisions based on the documentation output.
15. Select  **Cancel Generation** only if you need to stop the current generation process.
16. Review the generated technical documentation after processing is complete.
17. Select **Export Documentation** if you need to save or share the completed documentation outside Atloria.
18. Select **View Your Documentation** → when you are ready to open and review the generated documentation.

 *Screenshot pending: The source editor showing the Source and Technical view options and the language selector.*

Tips and Best Practices

- Provide representative source material. Analysis is more useful when the material includes the application areas you want documented, rather than isolated files with no surrounding context.
- Include related parts of the application together when possible. For example, documentation is stronger when it can connect a **COMPONENT** to the hook it uses, the service it depends on, or the page where it appears.
- Review both views before generation. Use **Source** to verify the material being analyzed, then use **Technical** to focus on the documentation-oriented result.
- Choose the correct language formatting in the source editor. Proper formatting makes source content easier for you to review before generation.
- Do not cancel generation unless necessary. When **4. Generation Running** is displayed, Atloria is actively processing your material. Canceling ends that generation attempt, so you may need to start again.
- Treat analysis categories as a coverage check. If your documentation should explain database behavior, API schemas, or infrastructure, make sure relevant source material is included alongside frontend and backend material.
- Use **Add code examples** selectively. Add examples that clarify important behavior, relationships, or implementation patterns. Avoid adding unrelated

samples that could make the generated explanation less focused.

- Review generated output before exporting. Use **View Your Documentation** → to confirm that key application entities and relationships are represented accurately.

Related Pages

- [Generate Technical Documentation](#)
- [Add Code Examples](#)
- [Import Existing Documentation](#)
- [Review Generated Documentation](#)
- [Export Documentation](#)
- [Manage Documentation Generation](#)

Troubleshooting

Generation does not start

Confirm that you are in **Technical Docs** and that source material or code examples are available for the project. Then select **Generate Documentation** → again. If you imported existing documentation instead, verify that the import is complete before starting generation.

The source view does not look correct

Select **View Source**, then choose **Source** in the toolbar. Check the language selector and select the language that best matches the source material you are reviewing. This improves readability while you validate the content.

Generation is taking longer than expected

Check whether **4. Generation Running** is still displayed. If it is, allow the process to continue. Use  **Cancel Generation** only when you intend to stop the current attempt; canceling does not produce completed documentation.

Generated documentation is missing expected application details

Review the supplied source material and add relevant examples with **Add code examples**. Include connected material where possible—for example, pages with their components, components with their hooks, or services with related repositories—then select **Generate Documentation** → to run generation again.

