

Understanding Parser Availability Before Code Upload

August 25, 2026

Checking what parser support means before you upload code

In Atloria, **parser availability** means the code parsing workspace can recognize the **programming language** you plan to upload and, when relevant, the **framework** used in that codebase. You usually confirm this before you move forward with **repository selection**, **source file upload**, or the action that starts parsing. If you skip this check, you can end up uploading code that Atloria cannot fully interpret.

A quick support check helps you answer two separate questions:

- Is the **language** listed as supported in the parser selection area?
- If the project depends on a specific **framework**, is that framework also available for the selected language?

Those two checks matter because **language support** and **framework support** are not always the same. Atloria may support a language in a general way, while only supporting certain frameworks built on that language. For example, you might see a language available in the list, but not see the framework your team actually uses. In that case, parsing may still be possible, but the results may be more limited than expected.

Use the parser selection screen as your decision point. Before you click the button that continues to upload or connect code, compare the options shown in Atloria with the stack you already identified for the project.

[SCREENSHOT: Parser selection screen showing language list and framework options]

Checking support first helps you avoid:

- **Failed parsing attempts**
- **Incomplete analysis results**
- **Choosing the wrong parser for the repository**

- **Rework during project onboarding**

If you already reviewed support coverage at a higher level, use that earlier planning work here rather than repeating it. For broader evaluation guidance, see [Using Language Support Information During Project Planning](#).

Gathering the language and framework details for your project

Before you open the upload flow in Atloria, collect the exact stack details for the codebase you want to parse. This is especially important when a repository includes more than one language or when different folders use different tools. The goal is to identify what Atloria should parse **first**, not to describe every technology used across the entire project.

Start by identifying the **primary language** in the part of the repository you plan to upload. In mixed repositories, the main language is usually the one used in the largest portion of source files or in the module you want documented first. If your team works in a monorepo, do not assume one language choice will fit every folder.

Then confirm the **framework or runtime** that shapes how the code is organized. This might be a web framework, mobile framework, or build-oriented framework. The framework matters because Atloria may show different parser options depending on the language you choose.

Useful places to verify this before upload include:

- **Project manifest files**
- **Dependency files**
- **Workspace or package files**
- **Folder structure and module names**
- **Existing project setup notes**

As you review those files, focus on what is actually present in the repository rather than what the team informally calls the project. A team may describe a codebase by a broad technology label, while the uploaded folder uses a different framework or no framework-specific structure at all.

If the repository contains several supported stacks, decide which one will be parsed in the first pass. That may mean narrowing the upload to:

- One **repository**

- One **module**
- One **service**
- One **folder path**

[SCREENSHOT: Project workspace showing upload area before parser selection]

This preparation makes the parser selection step much easier because you can match Atloria's labels to the real contents of the code you are about to upload, instead of guessing after the upload has already started.

Reviewing supported languages and frameworks in the parser selection interface

When you reach the parser selection step in Atloria, use the options on that screen as the final confirmation before upload. This is where you check whether the stack you identified earlier appears exactly as an available choice. Do not rely on memory or team shorthand here—use the names shown in the interface.

Look first for the **language list**. This list tells you which languages Atloria can currently parse in that workflow. Select the language that matches the code you plan to upload. After choosing it, check whether Atloria also shows a **framework field** or **framework selector**. If it does, review the available framework entries tied to that language.

As you compare your project stack with the screen, pay attention to support signals such as:

- **Enabled options** you can select
- **Unavailable entries** that cannot be chosen
- **Compatibility messages** shown after a language is selected
- Changes in the **framework list** when you switch languages

These details help you confirm whether Atloria supports the exact combination you need. A close match is not enough. If your team uses one framework and Atloria shows a different framework with a similar name, treat that as a separate option rather than assuming it will work the same way.

A careful review at this stage helps prevent two common mistakes:

- Choosing a supported **language** and assuming every framework under it is also supported

- Picking a framework that sounds familiar but does not match the repository you are uploading

[SCREENSHOT: Language dropdown expanded with a framework selector below it]

If the screen shows only a language choice and no framework field, use that as a sign to confirm whether your upload will rely on **language-only parsing**. If you need more context on how parser coverage affects results, refer to [Understanding Parser Coverage and Stack Compatibility](#) and [Evaluating Supported Languages and Framework Coverage](#).

Deciding how to proceed when your stack is fully supported, partially supported, or unavailable

After checking the parser selection screen in Atloria, place your project into one of three support situations: **fully supported**, **partially supported**, or **unavailable**. This gives you a clear basis for deciding whether to continue to upload or pause and adjust your plan.

When your stack is **fully supported**, both the **language** and the **framework** appear as available choices in the parser selection area. In that case, you can move ahead with confidence to the next step, such as selecting the repository or uploading source files. This is the best case for teams expecting framework-aware parsing results.

When your stack is **partially supported**, the **language** appears in the list but the **framework** does not. In this situation, you need to decide whether **generic language parsing** is acceptable for the first pass. That decision usually depends on what you need from the parsing results and whether missing framework-specific interpretation would affect your documentation scope.

When your stack is **unavailable**, neither the language nor the framework appears as a usable option. If that happens, stop before uploading code. Uploading anyway can create confusion, wasted setup time, and expectations that Atloria will produce results it cannot support for that stack.

Use these support states to guide planning conversations:

- **Fully supported**: proceed with normal onboarding and parsing expectations
- **Partially supported**: note possible limits in output and review scope carefully
- **Unavailable**: pause the upload plan and reconsider what codebase to parse first

[SCREENSHOT: Parser selection screen showing available and unavailable support states]

Support status also affects timelines. If parsing is central to project onboarding, a partially supported or unavailable stack may change what documentation can be generated first, which teams can review results, and what stakeholders should expect during setup. Treat this as a planning decision before any parsing job begins, not as something to sort out after upload.

Recording parser availability decisions for project planning

Once you confirm support in Atloria, write down the decision where your team tracks project setup details. This keeps future uploads consistent and helps other team members understand why a particular parser choice was made. The most useful notes are short, specific, and tied to the exact repository or folder being onboarded.

Record the following details from the parser selection step:

- The selected **language**
- The selected **framework**, if one was available and chosen
- The observed support status: **fully supported**, **partially supported**, or **unavailable**
- The repository, module, or folder included in the first parsing pass

If Atloria supports only the language and not the framework, note that clearly. That distinction matters later when someone revisits the project and assumes framework-aware parsing was used. A simple note such as “language-only parsing selected for first upload” can prevent confusion during review.

It also helps to document anything intentionally left out of the first upload, such as:

- Modules using a different stack
- Folders with unsupported frameworks
- Parts of a monorepo postponed to a later phase
- Repositories excluded from the initial onboarding scope

[SCREENSHOT: Project notes or planning document alongside parser selection choices]

When parser availability affects scope or schedule, include any approval details from the people overseeing project setup. In many teams, that means noting whether a **Project Administrator** or other decision-maker agreed to proceed with language-only parsing, delay certain modules, or narrow the onboarding target.

These notes become especially valuable when the project moves from setup into documentation work. They give reviewers context for why some reference coverage appears first and why other areas were intentionally deferred. For related onboarding decisions, see [Creating Projects and Completing Onboarding](#) and [Uploading and Parsing Code in the Workspace](#).

Fixing common parser availability checks before starting an upload

Most parser availability problems in Atloria come from checking the wrong stack, checking the wrong screen, or treating a mixed repository as one upload target. Before you start an upload, use the parser selection area to resolve these issues while changes are still easy to make.

If the **selected language does not appear** in the parser list, first confirm that you are on the correct upload or parser selection screen. Then verify the project's primary language again using the repository contents you plan to upload. In mixed codebases, users often choose the language of the overall product instead of the language used in the specific folder or module being parsed.

If the **framework option is missing after choosing a language**, do not assume Atloria failed to load it. Recheck whether that framework is actually used by the repository. Teams sometimes describe a project by its deployment tooling, hosting setup, or company standard, even when the uploaded code does not use that framework directly.

If the **repository contains multiple stacks and support looks inconsistent**, break the validation into smaller parts. Instead of treating the whole repository as one parser target, review support by:

- **Module**
- **Service**
- **Folder**
- **Repository segment planned for first upload**

This usually makes the parser choice much clearer.

If users are **ready to upload despite unclear support status**, pause the workflow. It is better to delay the upload than to start parsing with the wrong expectations. A quick confirmation in the parser selection screen can save time that would otherwise be spent reviewing weak or incomplete results.

[SCREENSHOT: Upload flow paused at parser selection while reviewing language and framework choices]

For additional help comparing support options before upload, use [Viewing Supported Languages and Parser Coverage](#) and [Evaluating Language Support Before Parsing Code](#).

Overview

Use this document when you are at the point in Atloria where a code upload is about to begin and you need to confirm whether parsing is likely to work for the stack you selected. The focus here is not broad research into every supported technology. Instead, it is the practical check that happens immediately before you continue from parser selection to repository connection or file upload.

The main idea is simple: confirm the exact **language** and, when relevant, the exact **framework** shown in Atloria before you upload code. That check helps you avoid starting a parsing workflow with the wrong assumptions.

This guide is most useful when:

- You are onboarding a new project into Atloria
- You are preparing the first parsing pass for a repository
- Your repository includes more than one language or framework
- Your team needs to decide between framework-aware parsing and language-only parsing
- Project scope depends on what Atloria can parse right now

The sections above walk through the key decisions in order:

- Understand what **parser availability** means in Atloria
- Gather the real stack details from the codebase
- Review the **language** and **framework** options shown in the parser selection interface
- Decide whether the stack is fully supported, partially supported, or unavailable
- Record that decision for project planning
- Fix common issues before clicking through to upload

If you need the broader planning context behind these checks, use the earlier Supported Languages documents rather than repeating that evaluation work here, especially [Using Language Support Information During Project Planning](#). From here,

the next practical step is usually the upload workflow itself in [Uploading and Parsing Code in the Workspace](#).

Prerequisites

Before using this parser availability check in Atloria, make sure you have enough project information to compare your codebase with the options shown in the parser selection screen. You do not need deep technical expertise, but you do need a clear understanding of what you are about to upload.

Have these items ready before you begin:

- Access to the **project workspace** where the code upload will start
- The repository, source files, or code location selected for the first parsing pass
- The project's primary **language**
- The project's **framework**, if one is relevant to how the code is structured
- Basic visibility into project files that confirm the stack, such as dependency or manifest files
- A decision-maker available if support status could change onboarding scope or schedule

It also helps if you already completed the earlier support review work covered in the previous documents in this section. In particular, you should already be familiar with:

- How Atloria presents supported languages
- How parser coverage differs from full stack compatibility
- How support information affects project planning

If you have not done that review yet, start with [Viewing Supported Languages and Parser Coverage](#) and [Evaluating Language and Framework Support in Atloria](#).

You are ready to use this guide when you can answer these questions before upload:

- What codebase, module, or folder are we parsing first?
- What language does that code actually use?
- Does that code rely on a specific framework?
- Are we prepared to proceed if Atloria offers language-only support?

With those details in hand, you can use the parser selection screen in Atloria as a reliable checkpoint before starting the upload.