

Validating Audience Specific Publishing Before Launch

August 25, 2026

Confirming the audience rules and permissions before you validate

Before you start launch checks in Atloria, open the audience area for the project and review every audience that affects the release. Use the audience list to confirm the **audience name** is clear and matches the group you expect to publish to. Then open each audience entry and review the membership setup carefully. If your team uses group-based or profile-based conditions to decide who belongs to an audience, make sure those conditions still match the people who should see the content at launch.

Next, confirm that your own account has the access needed to complete the review. You should be able to open the project workspace, update audience settings, review version visibility, and open the published documentation view for testing. If you can view content but cannot change audience settings or version access, stop here and have a project administrator or documentation manager complete the validation with you.

You should also confirm that the release version already exists before testing. Open the version you plan to launch and check its current visibility. Make sure the version is set the way your team intends to release it, whether that is broader public access or a more limited audience-based release. If the wrong version is selected, your audience checks will not reflect the actual launch experience.

Finally, make a short review list of all content that uses audience targeting. Include pages, sections in navigation, linked articles, and any content areas your team marked for a specific audience. This gives you a checklist for both sides of validation: what the intended audience should see, and what everyone else should not see.

[SCREENSHOT: Audience list showing audience names and membership settings next to a project version selected for review]

Setting up the target audience for the content you plan to publish

Use the project's audience settings to create a new audience or update an existing one for the content you are about to release. Start by opening the audience entry and checking the details that determine who belongs to it. Pay close attention to the audience name and any membership conditions so the label matches the actual readers. If the audience is meant for a customer tier, partner group, or internal team, the setup should reflect that clearly before you apply it to content.

Once the audience is ready, go to the documentation content you plan to publish. Open the page, article, section, or navigation item and look for the audience targeting control in the editor or content settings. Select the intended audience and save your changes. After saving, confirm that the content item shows an audience label or restriction indicator so you can tell at a glance that the targeting was applied.

Do not stop at the main page. Open related child pages and linked articles to make sure the same audience logic is applied where needed. A parent page that is restricted while a child page remains unrestricted can still expose content through navigation or direct links. Likewise, a restricted child page under an unrestricted section can create a confusing experience for readers who can see the menu item but not the content they expect.

As you review, check the navigation structure from the same editing area. Make sure only the right pages are grouped under the audience-targeted section. If a page should stay public, leave it outside the restricted branch. If it belongs to the targeted experience, apply the same audience setting before moving on.

[SCREENSHOT: Documentation editor or content settings showing an audience selection control and a saved restriction label on a page]

Checking version access for each audience before launch

Open the version workspace for the release you are preparing and review the access settings tied to that version. Your first check is simple: confirm that the selected version is the one you intend to launch. Then inspect the visibility and audience access rules attached to it. A page can be targeted correctly and still fail in the public experience if the version itself is available to the wrong audience or unavailable to the right one.

Look at the version status together with its access settings. If the version is in a release-ready state but still limited to the wrong audience, your launch check will produce misleading results. You want the version status and audience restrictions to support the same release plan. For example, if this version is meant for a specific

audience, that audience must be able to open it from the published documentation view, while excluded audiences should be blocked consistently.

Test this with real viewing scenarios. Use at least one account that should have access and one that should not. If your team uses a view-as or impersonation option during review, use that to switch between audience perspectives. Open the version directly, then try reaching it through the version selector if one is available. After that, use search and any saved direct links to confirm the same access rules apply everywhere.

What you are looking for is consistency. An allowed audience should be able to find and open the version from normal navigation paths. An excluded audience should not be able to reach it through the version selector, search results, or a copied page link. If one route behaves differently, treat that as a launch blocker and fix it before release.

[SCREENSHOT: Version settings screen with visibility options and audience-based access controls for a selected release]

Reviewing the documentation as each audience will see it

Open the published documentation view in Atloria and switch to the version you plan to launch. This is the most important part of pre-launch validation because it shows the experience readers will actually get, not just what authors see in the editor. Start with the target audience view first. If you are signed in as a matching user, or using your team's audience testing method, confirm that the correct version opens and the expected navigation appears.

Review the left navigation carefully. Check section names, page listings, and nested items to make sure audience-restricted content appears only where it should. Then open each targeted page and read it as a user would. Verify the **page title**, body content, callouts, downloads, attachments, and related links. If the page includes links to other articles, open those too and confirm they stay within the intended audience experience.

After reviewing the allowed audience, repeat the same checks as a non-member audience or public visitor. The goal is not only to hide the page body, but to remove clues that the restricted content exists. Restricted pages should disappear from navigation, should not appear as visible related links, and should not open successfully from a copied URL if the viewer is outside the audience.

This is also the point where you should compare what appears in menus with what appears inside pages. Sometimes a page is hidden correctly, but an in-page link or

attachment still points to restricted material. Open those links one by one and confirm they follow the same audience rules as the page itself.

If you already completed the checks in [Validating Audience Targeted Publishing Before Release](#), use that work here as your baseline and focus only on the final published reader experience.

[SCREENSHOT: Published documentation view with version selector, left navigation, and a targeted page open for audience review]

Comparing targeted and public views to catch launch-blocking gaps

A common pre-launch problem in Atloria is that content looks correct in the editor but behaves differently in the published view. To catch this, compare the internal authoring view side by side with the public documentation view for the same version and page. Start with a page that has audience targeting applied, then verify that the same title, content blocks, and navigation placement appear for the intended audience after publishing.

Pay special attention to mismatches between page-level targeting and version-level access. A page may be assigned to a specific audience, but if that audience cannot open the selected version, the page will effectively disappear in the public experience. The opposite problem can also happen: the version is visible more broadly than intended, while individual pages are only partially restricted. That creates a confusing release where readers can enter the version but encounter missing or inconsistent content.

Check shared assets during this comparison. Open images, downloadable files, and cross-links from both targeted pages and public pages. Make sure a public page does not expose restricted information through a file download, image caption, or related article link. If a page is safe but its linked asset is not, the release still needs correction.

Document every issue you find while comparing views. A simple tracking table helps your team fix the right thing quickly before launch.

Audience	Version	Page or location	Issue type	What happened
Intended audience	Selected release version	Targeted page	Missing content	Page appears in editor but not in published view

Audience	Version	Page or location	Issue type	What happened
Excluded audience	Selected release version	Direct page link	Incorrect access	Page opens even though it should be hidden
Public visitor	Selected release version	Related link or file	Content exposure	Linked asset reveals restricted information

[SCREENSHOT: Side-by-side comparison of an internal page view and the published page view for the same targeted content]

Fixing common audience publishing validation issues

If an audience can open the version but cannot see the page, start with the page itself. Open the page settings and confirm the correct audience is assigned. Then check whether the page is included in the published navigation. In Atloria, a page can be properly restricted but still seem missing if it is not placed in the visible navigation structure for that version.

If a page is hidden in navigation but still opens by direct link, review both the page restriction and the version access settings. These two controls need to work together. Hiding a page from menus is not enough if the version remains open in a way that still allows the page URL to load for excluded readers. Recheck the version's audience access and test the direct link again with a non-member account.

If public visitors can still see restricted content blocks, inspect more than the parent page. Open any embedded sections, reusable content, linked articles, attachments, or shared assets used on that page. Sometimes the main page is targeted correctly, but a nested content block or linked file remains available more broadly and exposes information you meant to limit.

If the expected audience does not appear during testing, verify the test account itself. Open the audience definition and compare its membership rules with the user profile or group membership of the account you are using. If your team recently changed audience membership, save those updates fully before retesting. Then sign out and back in, or switch back to the published view again, so you are checking the latest audience state.

When an issue keeps repeating, return to the earlier setup guidance in [Publishing Documentation for Specific Audiences](#) and [Validating Audience Targeted Publishing](#) to confirm the targeting model is still aligned with the release you are preparing.

Overview

This guide focuses on the final validation pass for audience-targeted publishing in Atloria before launch. At this stage, you are not planning the audience model from scratch and you are not doing an early draft review. Instead, you are confirming that the release version, audience rules, page targeting, and published reader experience all match each other.

The core workflow has four parts:

1. Confirm the audience definitions and your own permissions.
2. Apply or verify audience targeting on the content being released.
3. Check version access for allowed and excluded audiences.
4. Review the published documentation view exactly as each audience will see it.

This document is especially useful when a release includes pages that should only appear for certain readers, such as customer-specific guidance, internal-only notes, or restricted release content. In those cases, it is not enough to verify the editor setup. You also need to confirm that navigation, direct links, related content, and version access all behave correctly in the published view.

Use this guide after you have already completed the broader pre-release checks in [Validating Audience Targeted Publishing Before Release](#). Here, the focus is narrower: catching launch-blocking gaps that only show up when the selected version is viewed through audience rules in the public documentation experience.

Prerequisites

Before you work through this validation in Atloria, make sure these items are already in place:

- You can sign in to Atloria and open the project workspace.
- You have access to the project's audience settings, version settings, and published documentation view.
- The documentation version you plan to launch has already been created.
- The pages, sections, or navigation items that use audience targeting have already been identified.
- You have at least one way to test an allowed audience and one excluded audience, such as separate user accounts or your team's review method for switching audience perspective.

- The content itself is already prepared for release, including page titles, links, callouts, and attachments.

It also helps to have these earlier guides completed so you are not repeating setup work during launch validation:

- [Defining Audiences for Targeted Documentation](#)
- [Managing Project Audiences for Targeted Documentation](#)
- [Validating Audience Targeted Publishing Before Release](#)

After you finish the checks in this guide, continue with [Validating Audience Specific Release Views](#) to review the final audience-facing release presentation in more detail.