Critical information about the package has been retained at the top of the page, along with information that all users prioritize—download numbers and usage metrics. These are the primary indicators of trust in a package: that "many eyes" have seen and used the package. For most developers, this is enough to justify using a package.

For packages that are less popular or less established, users cannot rely on this kind of "big numbers" information. In these cases, they look for authoritative information from trusted sources such as the registry website, authority bodies like software foundations, or maintainers with a reputable name.

Some users not only rely on the aforementioned information, but also personally check security details such as matching hashes and keys, reviewing source code, or checking provenance, or attestation pathways. These users typically aim to avoid risks such as introducing vulnerabilities that could expose user data (like financial or other private information), or put systems —such as company or university infrastructure—at risk.

From our research, only certain users understood what a 256 checksum and similar information are for, or how to use them. Therefore, placing attestation UI elements related to more complex or less commonly known security terms or processes can help users better understand these difficult concepts. By positioning the UI above the checksum, users are less likely to "check out" and stop paying attention to the page when they encounter unfamiliar terms they do not recognize or have time to investigate.

In this example from npm, you can see that the package exposes a lot of README information that the maintainers have worked hard on. During testing, users found this reasonable but somewhat over-engineered. They had difficulty finding the basic information they needed and relied heavily on the sidebar for the critical details required to decide whether to use the package.

Based on these findings, we believe that for package pages like this, presenting attestation information in the sidebar is best. Sidebar options are discussed in more detail later in the style guide.

The first place attestation information can be added is above the license information. Since the license is also detailed in the sidebar, users are generally comfortable with attestation content being positioned as high as possible on these pages. However, be aware that the likelihood of users discovering attestations above the license is low. Most users will not scroll through these long README pages unless they are specifically searching for security, provenance, or attestation details they cannot find elsewhere on the page or in a different tab. More commonly, users will go to the project repository to find information such as build logs, commit files, source code, and so on.

The first place attestation information can be added is above the license information. Since the license is also detailed in the sidebar, users are generally comfortable with attestation content being positioned as high as possible on these pages. However, be aware that the likelihood of users discovering attestations above the license is low. Most users will not scroll through these long README pages unless they are specifically looking for security, provenance, or attestation information that they cannot find elsewhere on the page or in a different tab. More commonly, users will go to the project repository to find information such as build logs, commit files, source code, and so on.

On this page (and others), you may find that detailed information is repeated in areas we suggest as supporting the attestation statement. From our user research and UI testing, we discovered that users did not find this confusing or off-putting. Instead, users were encouraged by "duplicate" information and had no problem with encountering the same thing in multiple places or seeing documentation in more than one section.

The researchers' hypothesis is that users seek information in a compartmentalized way. When users are specifically looking for attestation or security-related information about package builds, they tend not to notice other information not labeled as such, unless they deliberately switch their focus.

When users did notice that the same or similar information appeared in two sections or locations, they felt more confident in the package page's diligence at repeating critical information that might otherwise be missed. Their trust was also reinforced when two matching links appeared in different places. This response was connected to a general distrust of package information pages on registries. As one user stated, "They can put whatever they want on here, can't they? Who even approves and checks this? Anyone can put stuff on here." Seeing the same repeated in two places gave some users more confidence—if a source commit is linked in area 1 and area 2 and they match, it suggests someone is ensuring consistency.

Therefore, repeating information should not be a concern for platforms implementing this approach, but it is critical that the same headings, terms, labels, and destination links are used consistently across different locations

Here, you can see that the separated elements can either appear in the sidebar, in a dedicated section, or in a section, or they can take up only the minimum amount of space needed by the expandable sections to display additional information.

You'll also notice that this information is placed underneath the project description. This positioning risks users misinterpreting the information as belonging to the previous heading (in this case, Code of Conduct). In these situations, we recommend using a heading such as "Verification," "Provenance," "Project Verification," or "Secure" heading of your choice. We also recommend, specifically for PyPI, that some information typically found in the Download Files section is difficult for unfamiliar users (and even some regular visitors) to find. If the attestation information also included any SHA/checksum details and additional information related to these topics, then placing this section at the bottom of the project description would be less surprising.

In contrast to the comments in the above PyPI mock-up, when expanded, this section maintains its own relevance much more strongly and may not require moving information from the download files section.