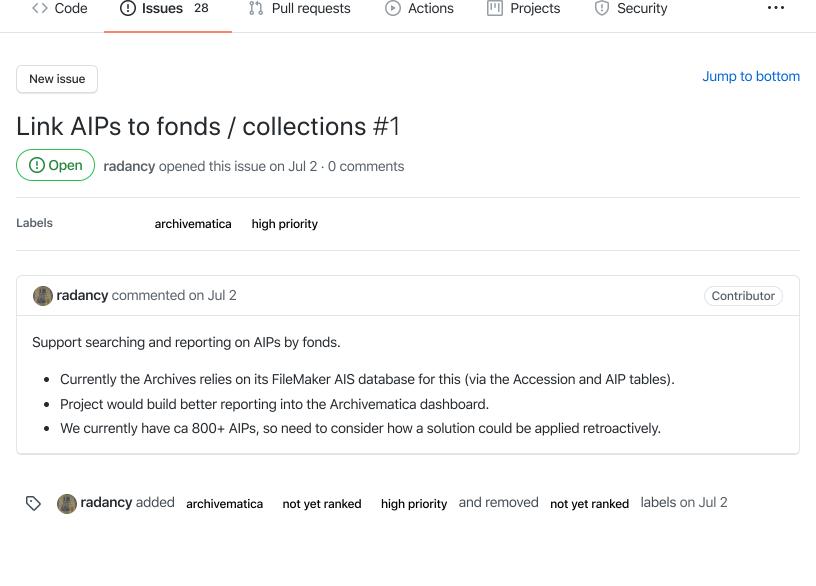
Software Development Priorities

Date: November 12, 2020

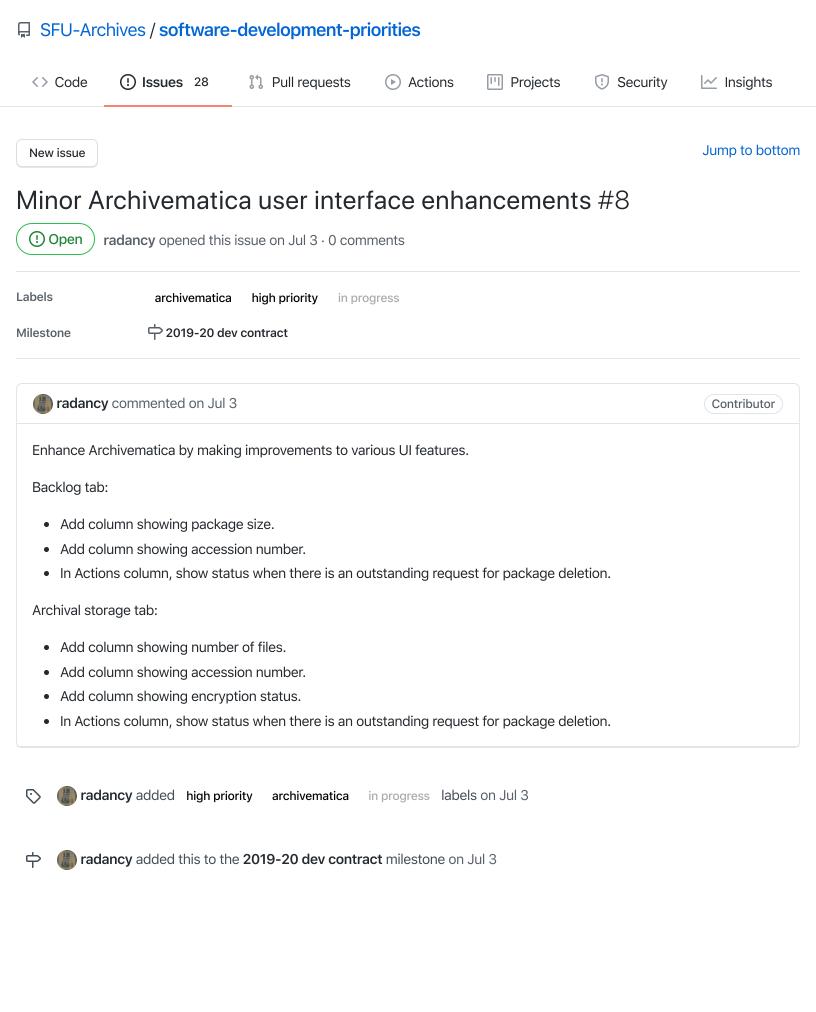
Status: Draft for review meeting

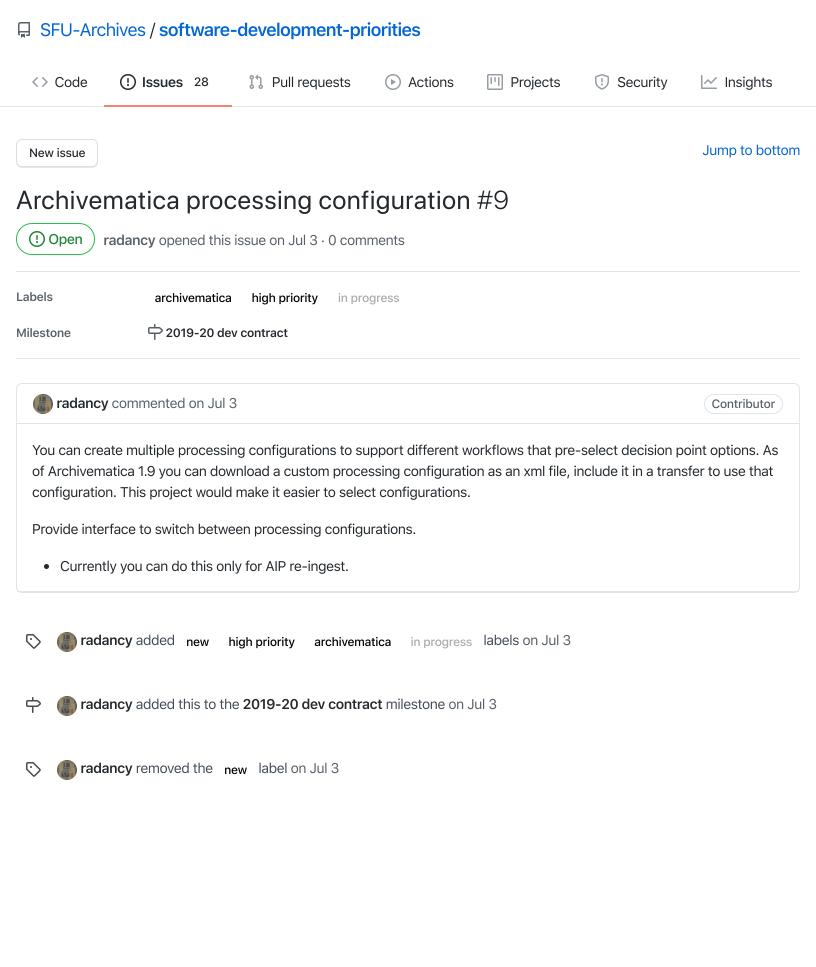


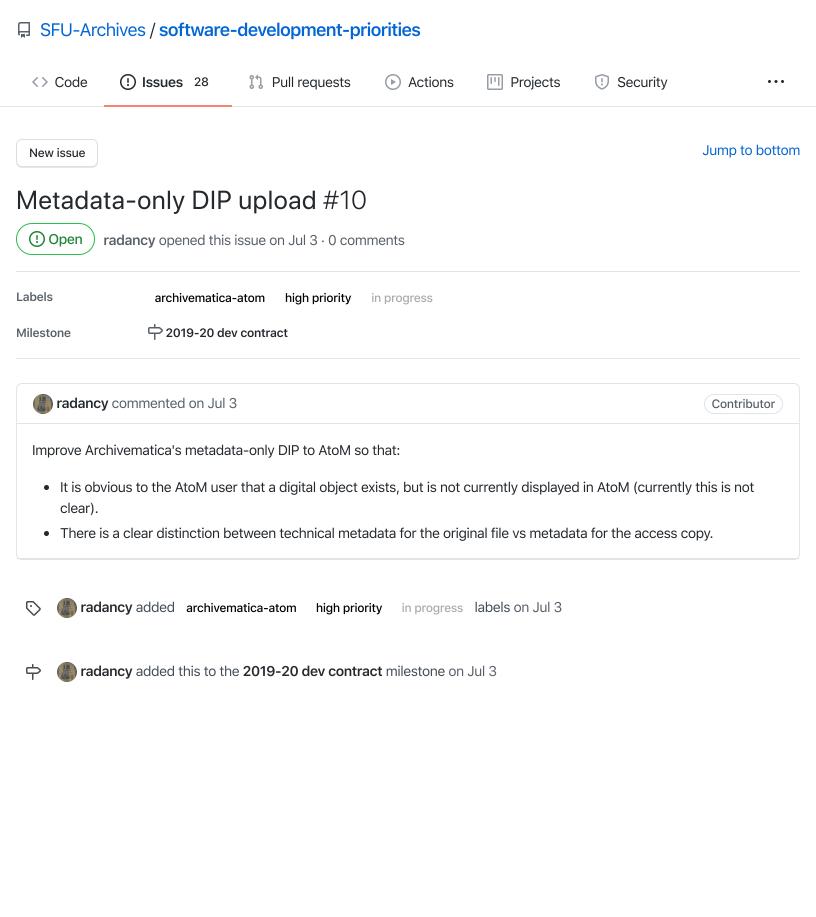
HIGH PRIORITY

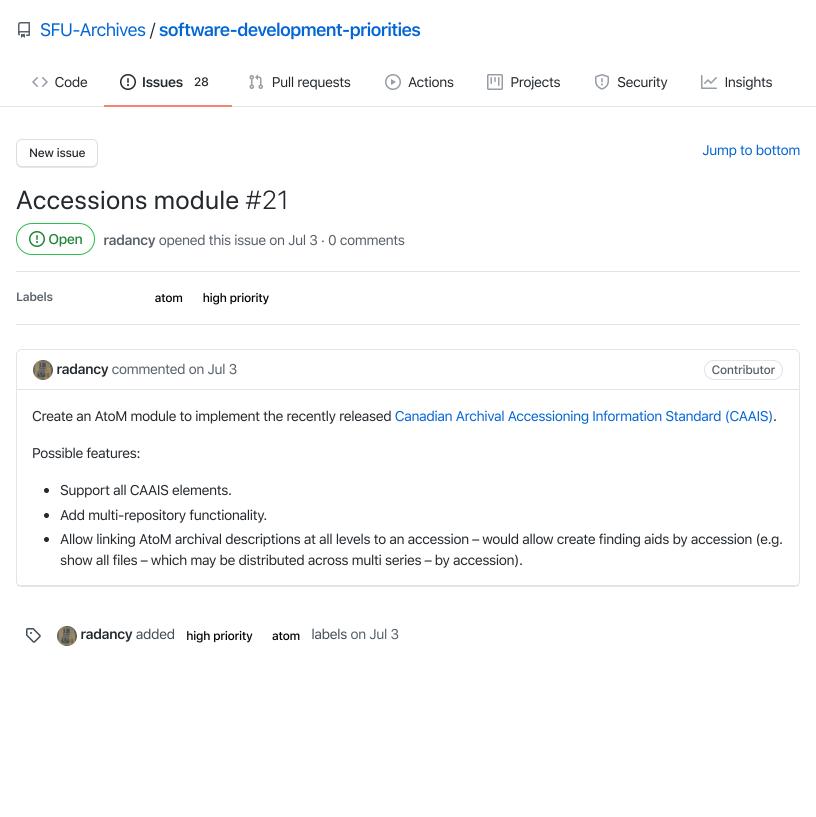


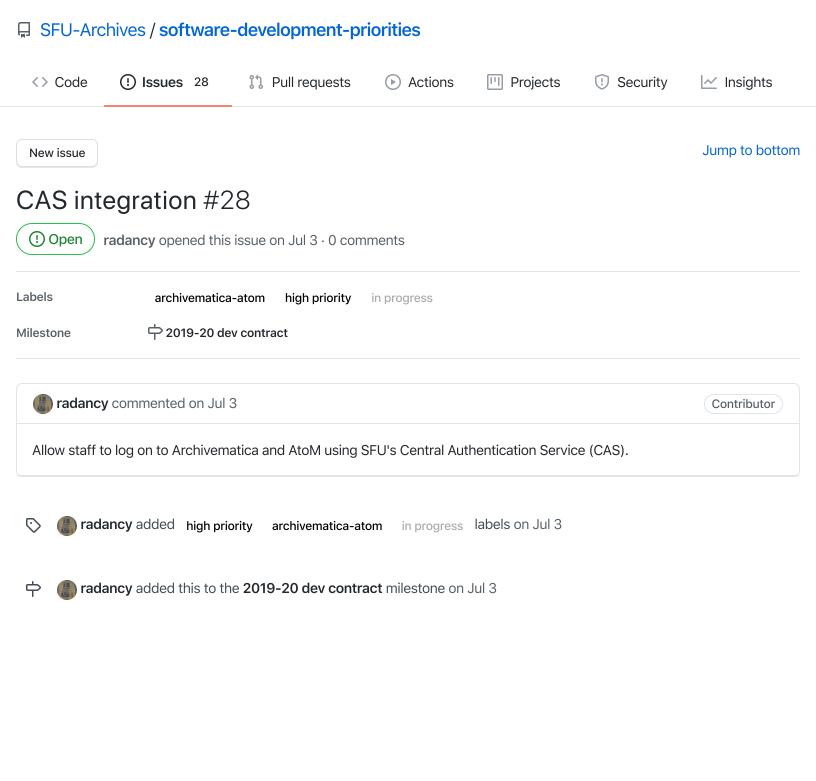
SFU-Archives / software-development-priorities



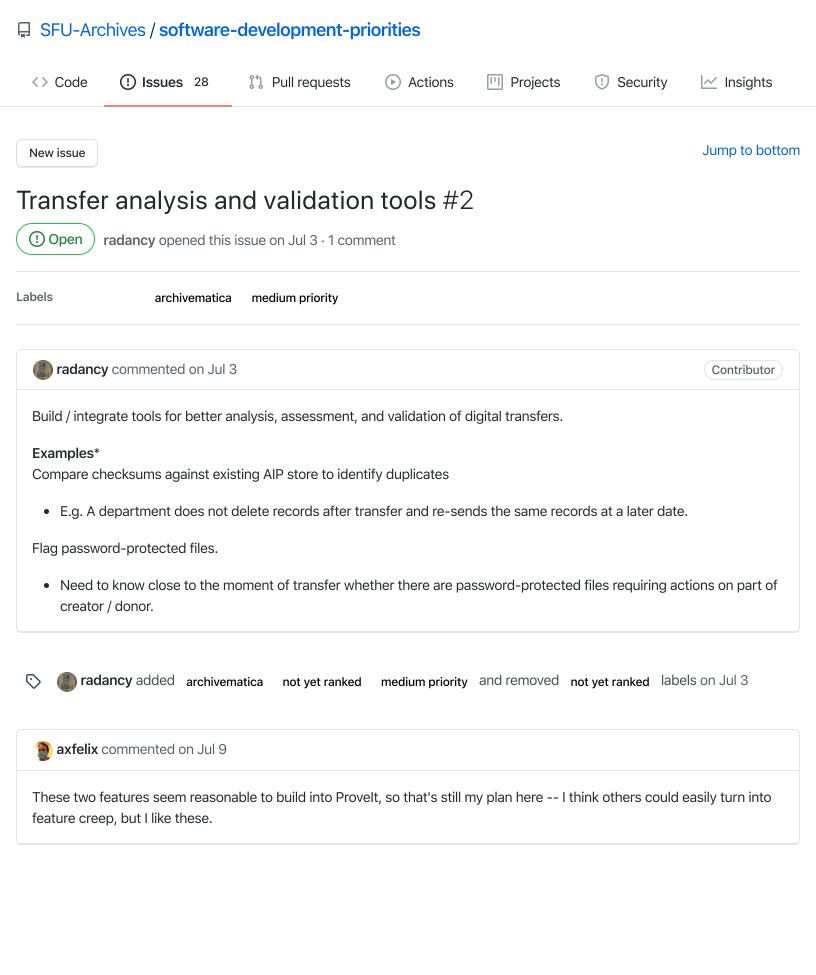


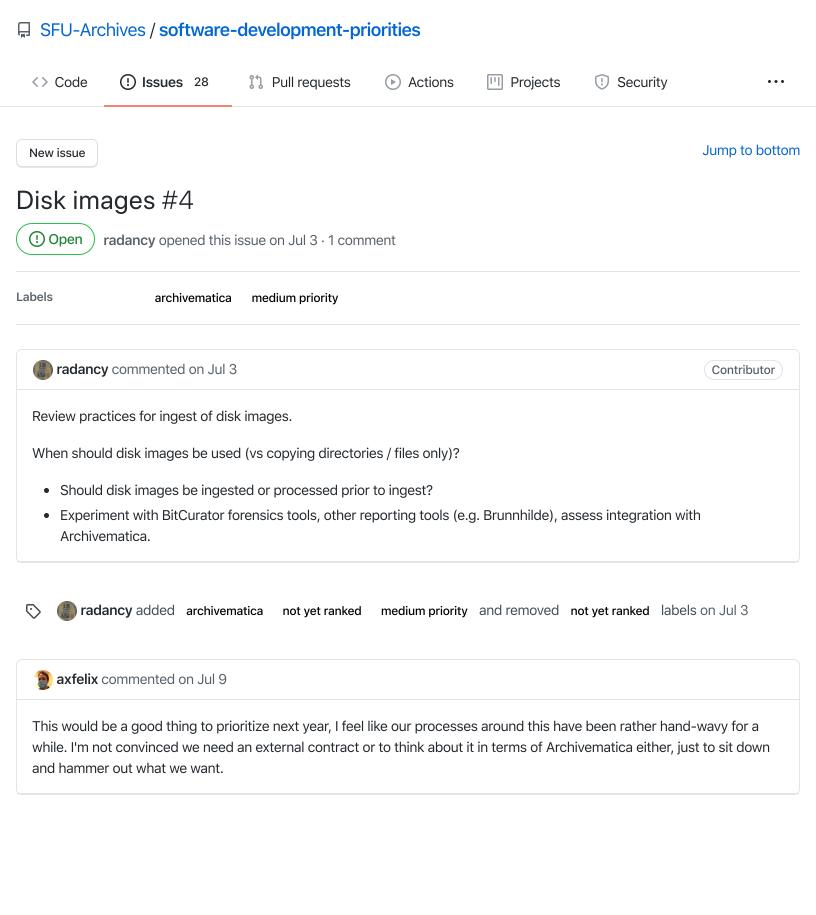




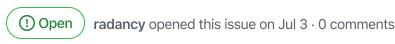


MEDIUM PRIORITY





Archivematica error handling #7



Labels archivematica medium priority

nadancy commented on Jul 3

Contributor

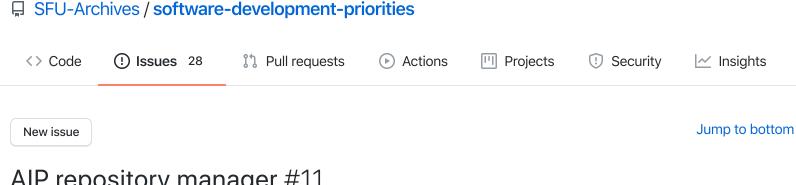
Archivematica microservices sometimes fails in non-critical ways (e.g. a JHOVE partial pass is represented as a fail in Archivematica); currently fails are handled ad hoc and without documentation. This project would establish a policy for consistently handling fails and build functionality in Archivematica to support the practice.

Policy, e.g.

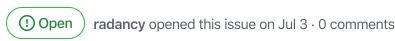
- Establish criteria for distinguishing critical vs non-critical fails.
- Specify actions for both cases.
- Indicate how decisions are documented and where.

Archivematica, e.g.

- Allow printing / export of identification fail reports.
- For fully automated ingest (using configuration files), generate a report on the conclusion of the ingest so archivist can review the ingest
- Set order of preference for format identification tools so Archivematica will run another tool when the preferred tool hits a fail.
- Allow using different tools on specific files without having to re-run the whole transfer.
- Provide interface for recording decisions to accept identification fails (included in METS file as PREMIS event).
- radancy added medium priority archivematica labels on Jul 3



AIP repository manager #11



Labels archivematica medium priority

radancy commented on Jul 3

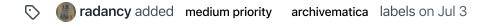
Contributor

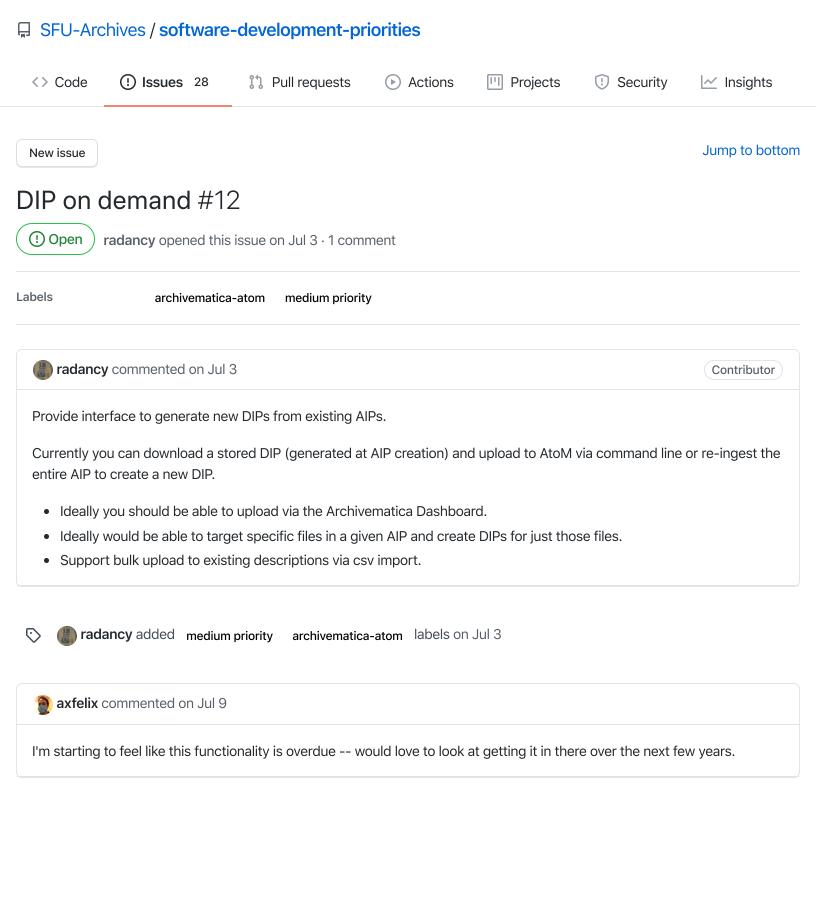
Develop repository management functionality for managing objects in the AIP store.

 E.g. reporting, searching, checking checksums, bulk re-normalization based on changes to the FPR, backup and recovery.

First phase could be a research project to determine the best model. Options might include:

- Enhance the Archivematica Dashboard (e.g. make some of the analysis tools from the Appraisal tab also available on the Archival storage tab).
- Enhance Storage Service (e.g. add management functionality).
- Build on existing work with Binder (make it fully open source).
- Build tools to search / report across METS files in the AIP store.
- Build integration with an existing repository management application (e.g. D-Space, Islandora)
- Build the manager as a stand-alone module separate from Archivematica pipelines.





SFU-Archives / software-development-priorities Code ① Issues 28 Pull requests Actions Projects ① Security Insights New issue Jump to bottom Email #13 ① Open radancy opened this issue on Jul 3 · 0 comments



medium priority

Build on previous work on email to develop solutions for transfer, arrangement and description, and access to email records.

Contributor

The project would involve research and policy as well as software development. Possible elements:

- Review scripts that managed export of email from Zimbra and convert to mbox do these still work with Outlook?
- Extract descriptive metadata from mbox DIP, map to AtoM description fields.
- Integrate email viewer in AtoM.

radancy commented on Jul 3

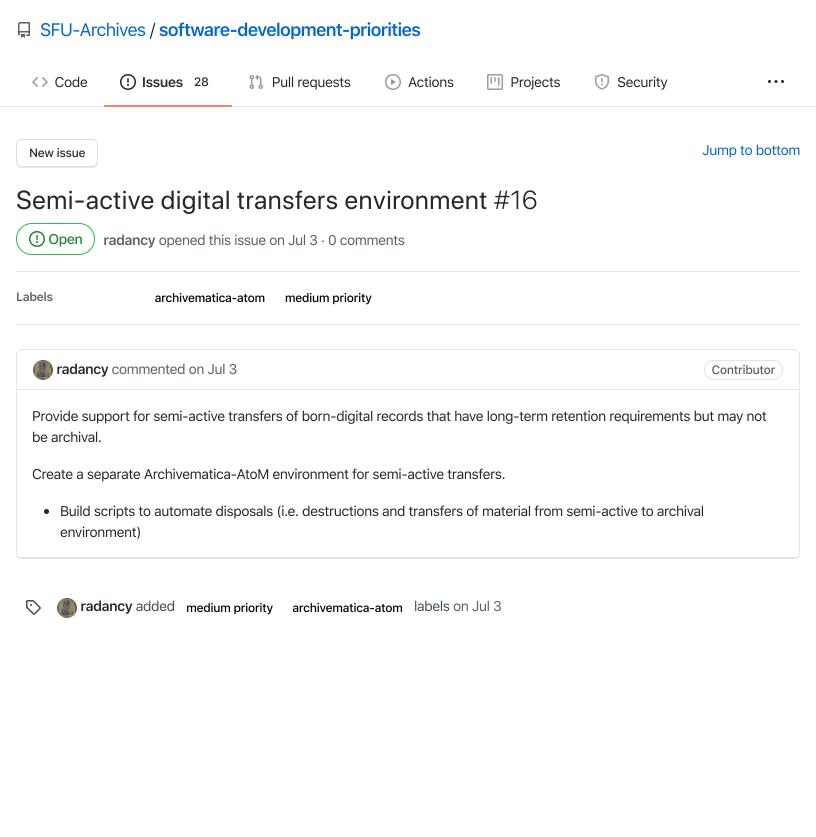
Labels

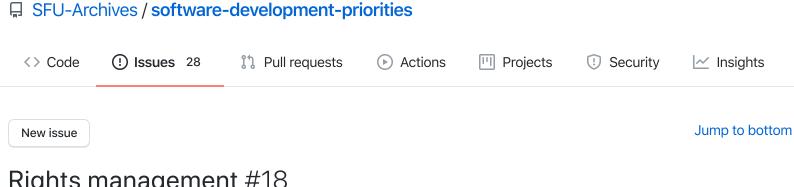
Configure IMAP as a transfer source in Storage Service.

archivematica-atom

Test ePADD to manage restrictions and access.







Rights management #18

① Open radancy opened this issue on Jul 3 · 0 comments

Labels atom medium priority

radancy commented on Jul 3

Contributor

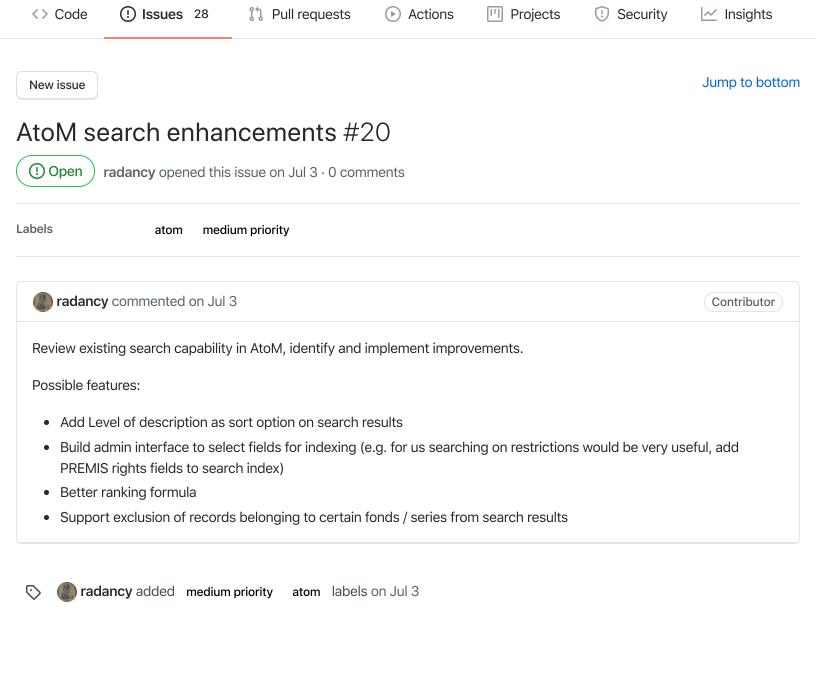
Build on the existing PREMIS rights area in AtoM to support types of conditional access, each with their own notice screen.

Current functionality:

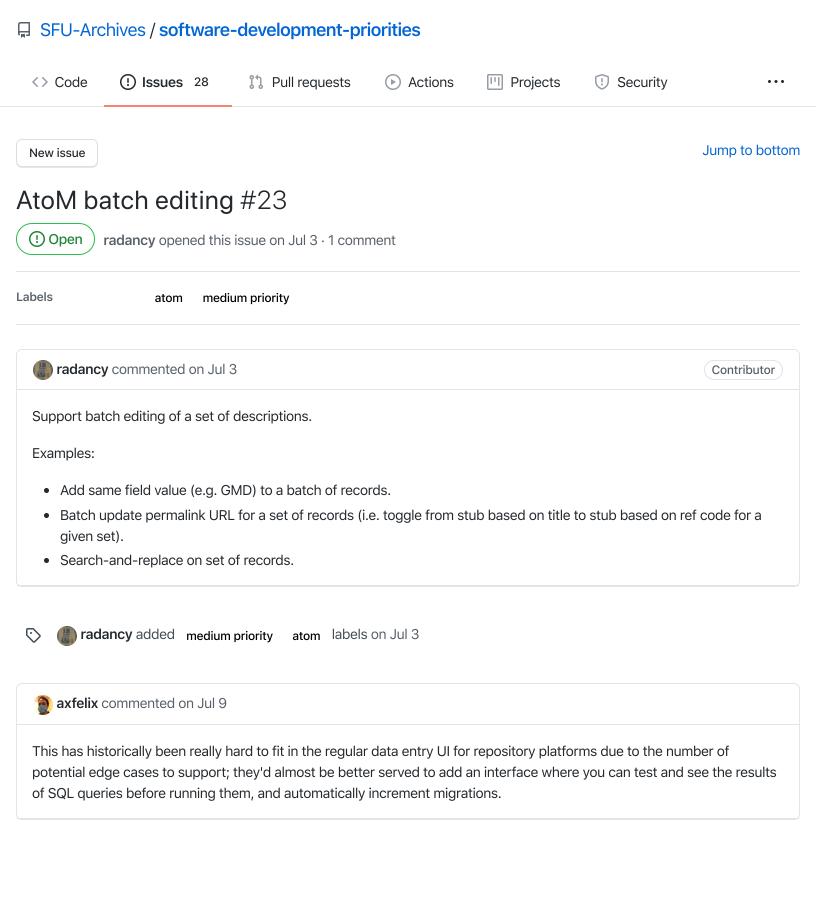
- You can set different access rules for different rights bases (statute, copyright etc).
- There are 3 restriction-states for each basis (allow, disallow, conditional).
- A popup notice screen for exists for one use case, where basis = "copyright" and restriction-state = "conditional".

Possible development:

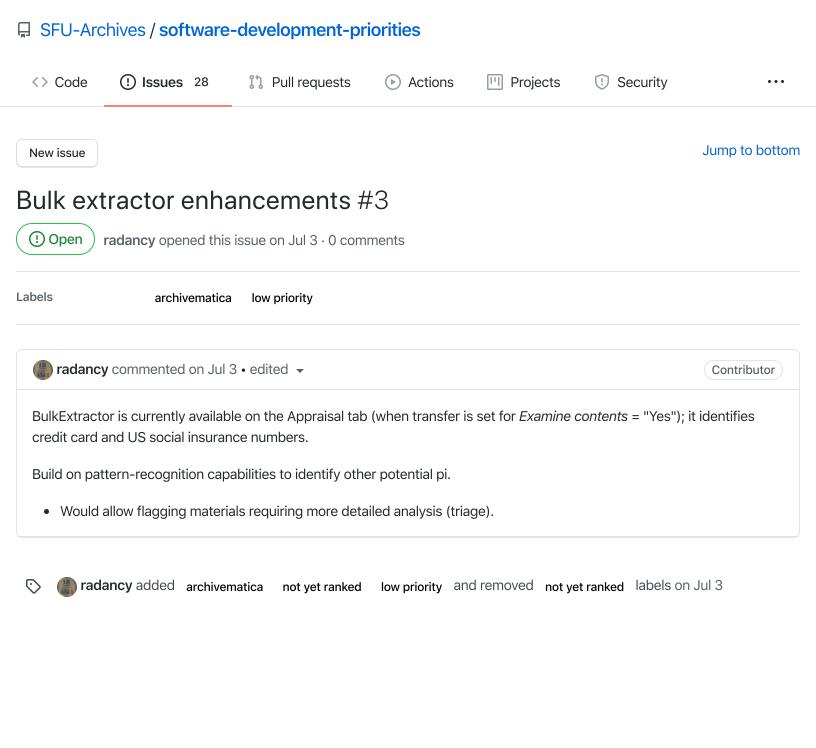
- Allow each rights basis to have its own conditional pop-up notice (rather than hard-coded to copyright).
- Integrate CC license icons
- Integrate standardized rights statements (from RightsStatements.org).
- radancy added medium priority atom labels on Jul 3

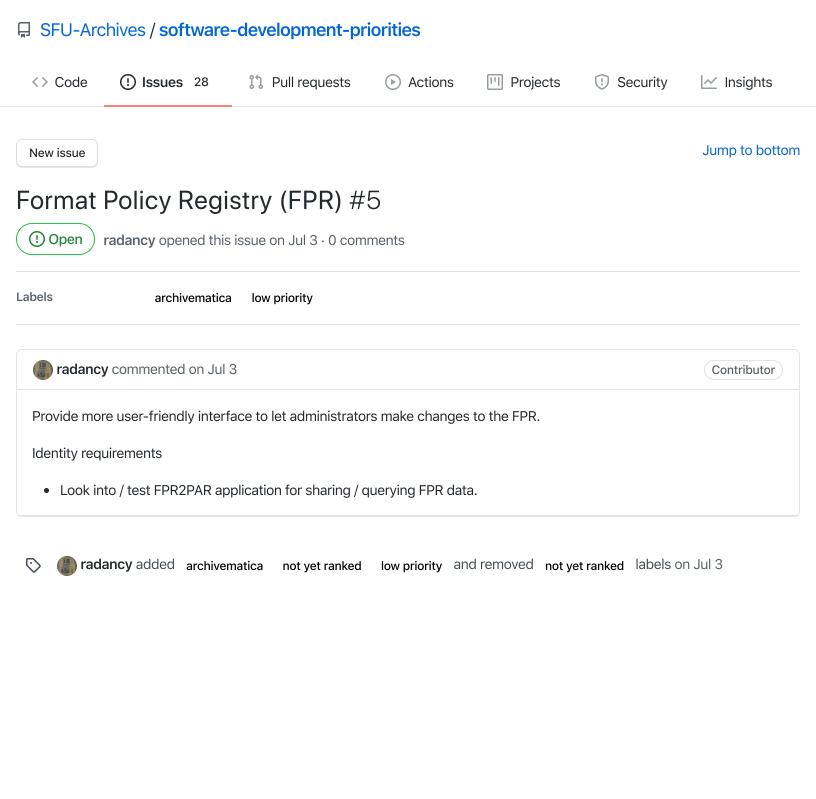


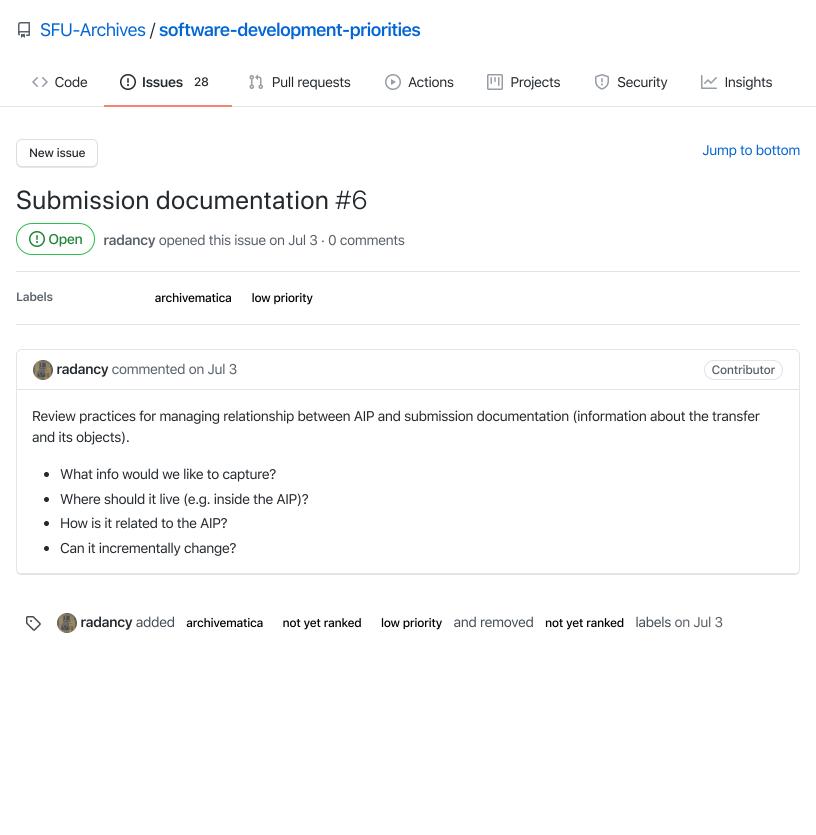
SFU-Archives / software-development-priorities

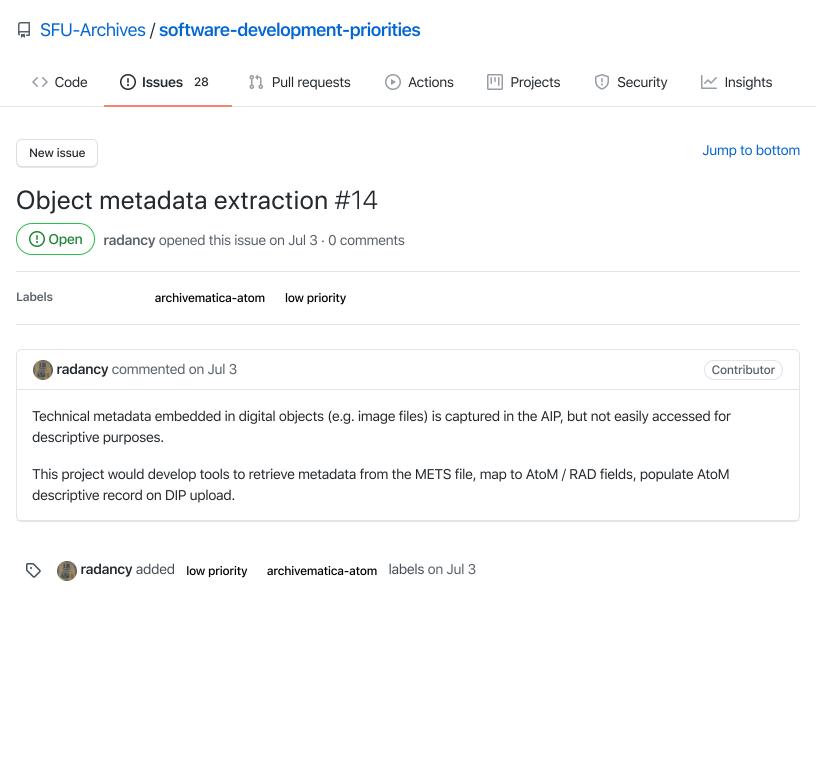


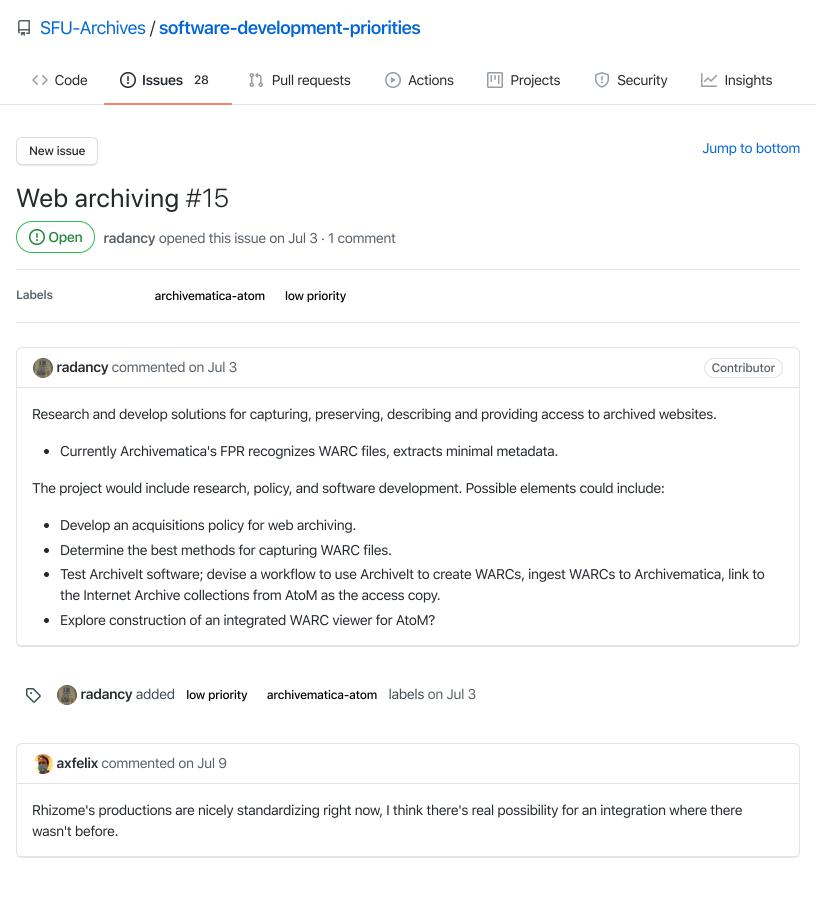
LOW PRIORITY

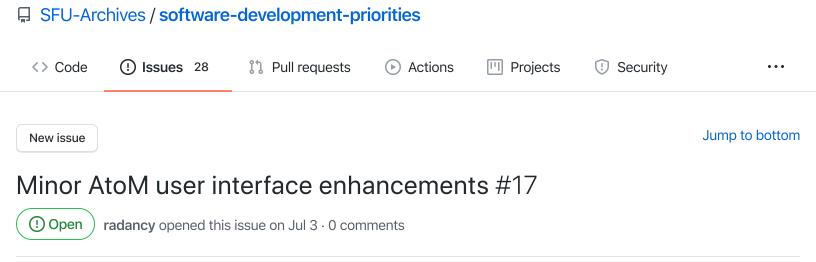












radancy commented on Jul 3 • edited ▼

low priority

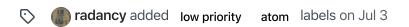
atom

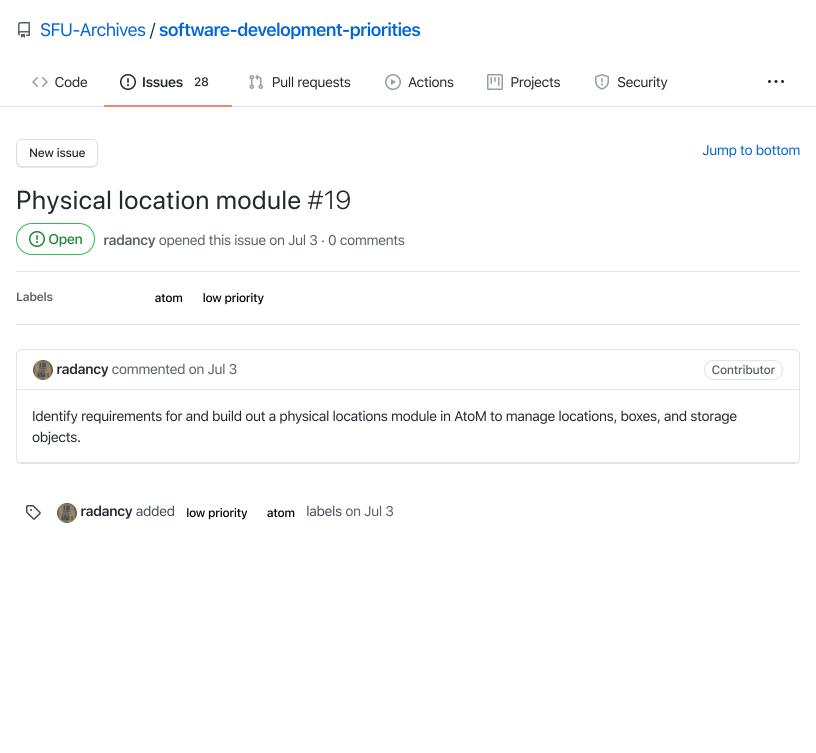
Labels

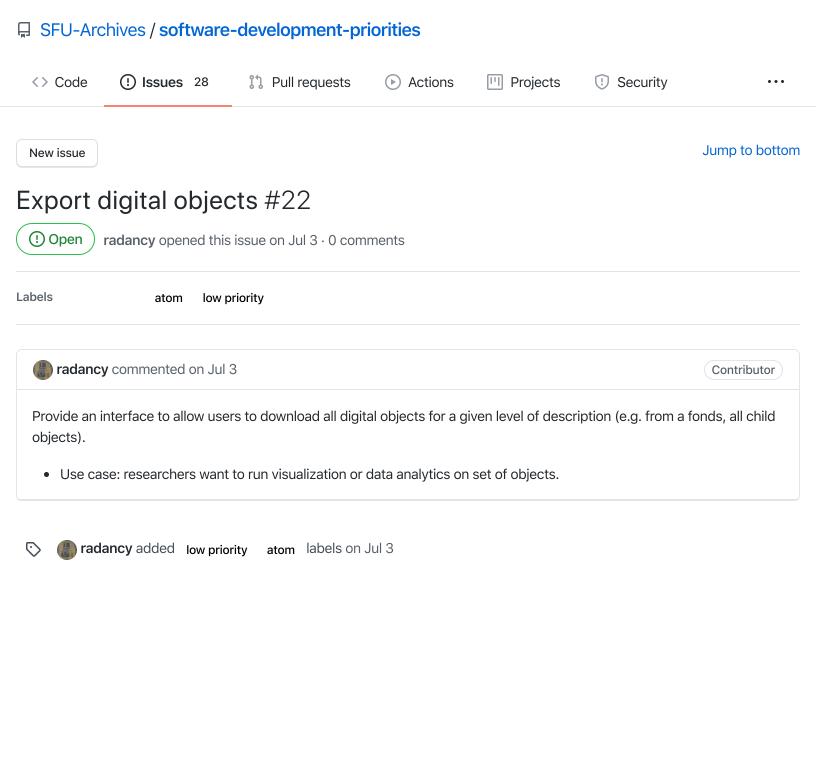
Contributor

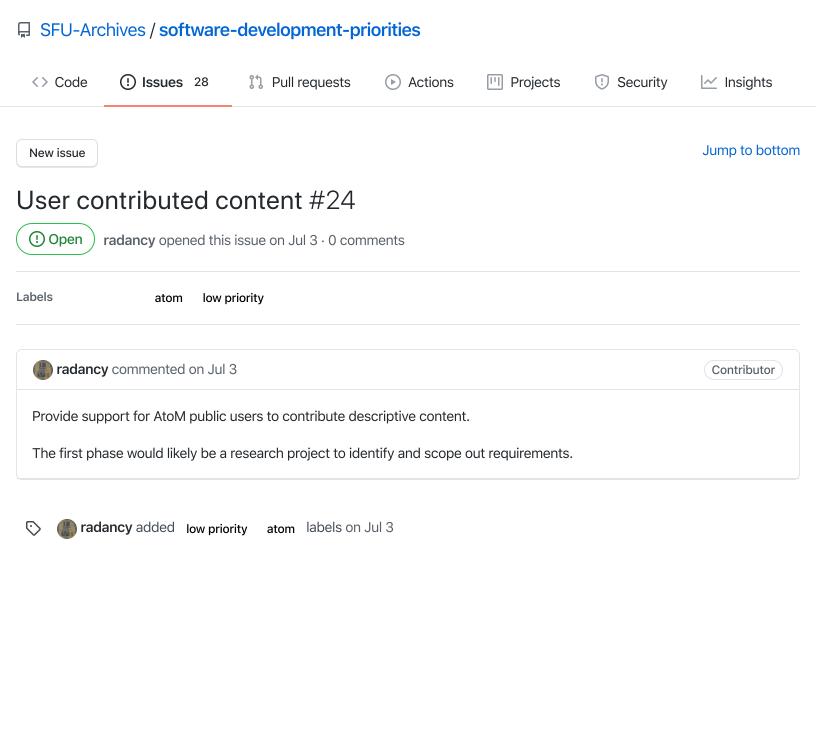
Identify and implement a set of relatively minor user interface enhancements. Possible features:

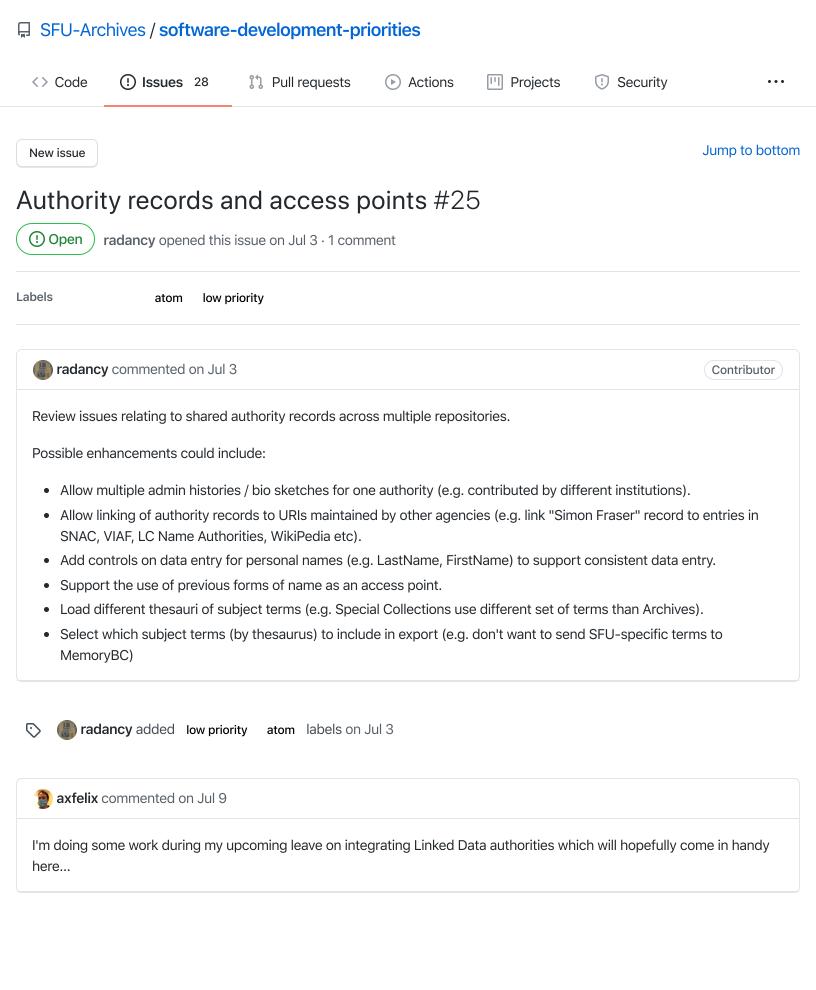
- Make treeview sticky: when you save a record, treeview reverts to top of tree, losing view of current record's position in the hierarchy).
- Add a "recently added" widget to the AtoM Home page to display new content; and provide an admin interface to manage content that appears.
- Support a rotating set of images for the AtoM Home page; provide an admin interface to allow easy upload / deletion.
- Restructure Related materials, Associated materials, and Sources fields as repeatable sets (Identifier, Note, URL); for Associated materials the URL can point to holdings in other repositories.
- Expose AtoM version number on the Home page (right now only an administrator can access; can be useful to know for liaising with AABC and MemoryBC).



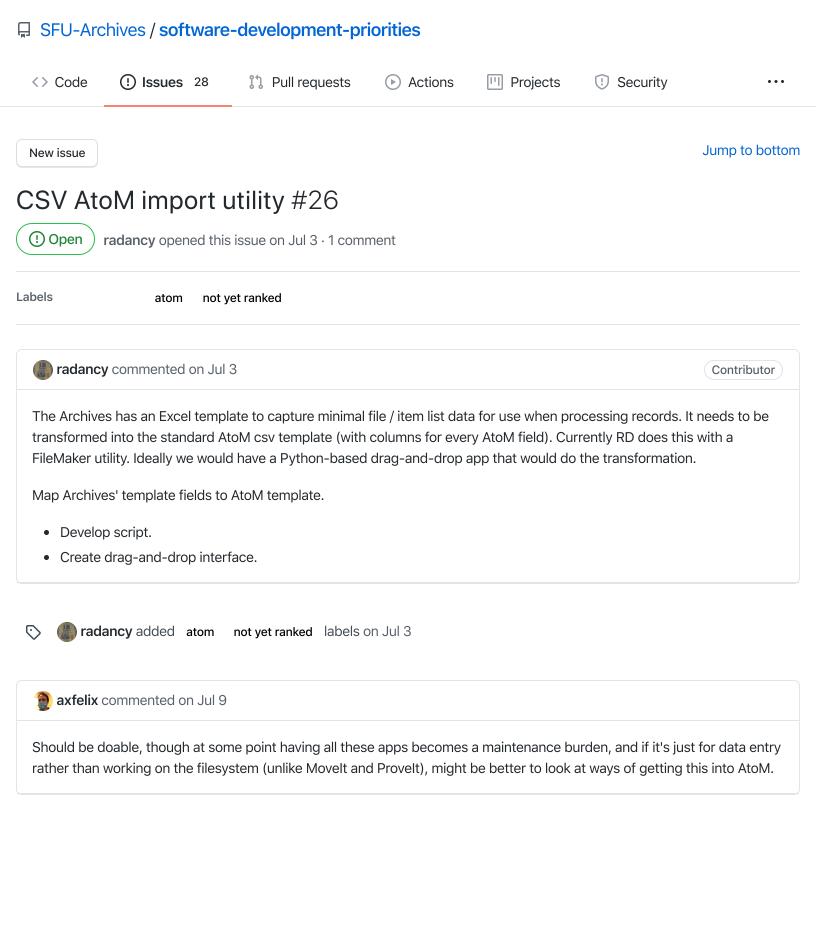








NOT YET RANKED



SFU-Archives / software-development-priorities <> Code (!) **Issues** 28 11 Pull requests Actions Security Projects Jump to bottom New issue Selective de-indexing of digital objects #27 ① Open radancy opened this issue on Jul 3 · 3 comments Labels atom not yet ranked radancy commented on Jul 3 • edited 🔻 Contributor Support ability to remove indexing from selected digital objects for privacy reasons. Use case: individual requests removal of name from public document (e.g. Convocation Book). Record would remain available in AtoM, but individual's name would not be picked up in searches. radancy added atom not yet ranked labels on Jul 3 🗿 axfelix commented on Jul 9 I'm guessing this would require setting a database flag on the objects that's editable from the interface and gets ignored by all the indexing calls -- should be very doable within the existing codebase with a small-to-medium amount of effort. 🗿 axfelix commented on Jul 9 Would be easier if we don't need to worry about pre-setting this permission, as then we would only have to worry about handling it on objects that already exist in AtoM, rather than on ingest or communication from Archivematica. axfelix commented on Aug 31 Adding this after the page.goto(); block seems to work: await page.evaluate((sel) => { var elements = document.querySelectorAll(sel); for(var i=0; i< elements.length; i++){</pre>

```
elements[i].parentNode.removeChild(elements[i]);
}
}, "signup-prompt")
```