2020-11-12 DSpace 7 Working Group Meeting

Date

12 Nov 2020 from 15:00-16:00 UTC

Location: https://lyrasis.zoom.us/my/dspace (Meeting ID: 502 527 3040). **Passcode**: dspace

- More connection options available at DSpace Meeting Room

Beta 5 Sprint : Ongoing

ⓘ

- Ongoing development on Beta 5
- View PRs assigned to you for review/testing: https://github.com/pulls/review-requested

Agenda

- **(BEFORE MEETING IN #dev-sprint)** Developer Stand Up - Developers give brief updates on their effort (or their team's effort).
  - Update/see "Current Work" section below based on your status. *Please feel free to update prior to meeting.*
  - Please highlight any new work (needing reviews/testing), any blockers (for you), and any discussion topics you may have.
- (30 mins) General Discussion Topics
  1. *(10 mins)* Exception messages from a failed REST request are not properly displayed in the response: https://github.com/DSpace/DSpace/issues/3045 (It seems the problem is not clear from the description yet)
     a. Tim created a related PR to disable the stacktrace from error responses: https://github.com/DSpace/DSpace/pull/3048
  2. *(10 mins)* Start discussion of "150 GET requests on Community & Collection page" (would require REST API changes): https://github.com/DSpace/dspace-angular/issues/918
  3. *(10 mins)* Start discussion of "Entity type/label as part of URL path" (requested by Google Scholar): https://github.com/DSpace/dspace-angular/issues/745
- (30 mins) Planning for next week
  - Review of our Beta 5 Project Board & assigning PRs to reviewers.
  - Beginning to assign work

Attendees

- Art Lowel (Atmire)
- Andrea Bollini (4Science)
- Tim Donohue
- Lieven Droogmans
- Giuseppe Digilio (4Science)
- Ben Bosman
- Paulo Graça

7.0 Release Goals

These resources define the prioritization and general schedule we are working towards

- DSpace 7 Release Goals : overview of goals/timelines & beta release process
- DSpace 7 Project Boards : our planning/scheduled boards which details which features are scheduled for each Beta release.

Current Work

Project Board

DSpace 7.0 Beta 5 Project Board: https://github.com/orgs/DSpace/projects/4

To quickly find PRs assigned to you for review, visit https://github.com/pulls/review-requested (This is also available in the GitHub header under "Pull Requests Review Requests"

Security / Performance Tests

Brainstorming options for security testing & performance testing. How do we want to handle both of these prior to 7.0 final?

1. Security Review/Scanning of pre-7.0: See DSpace 7 Security Analysis
2. Performance testing of pre-7.0: See DSpace 7 Performance Analysis

Delayed / Needs Discussion

1. Finalize / approve the initial list of all authorization features which we should implement for the /api/authz/features REST endpoint. This list of features should be limited to *only features which are required to enable/disable User Interface functionality.* (In other words, we can always add more features in the future. We just need to approve the list necessary for 7.0)
   i. Review current spreadsheet (from Andrea Bollini (4Science)) : https://docs.google.com/spreadsheets/d/1182LcD_WqIZRbUGWpLtBw0aOMR9jhbOVB7GZqtTpR9A/edit?usp=sharing
      1. Art Lowel (Atmire) : I don't see any immediate issues with the current set of features, but I would prefer a consistent naming scheme. I'd use canDoSomething for everything
      2. Tim Donohue added possible renames of these features based on Art's idea (see cell comments in spreadsheet). I like the "can[DoSomething]" naming scheme as well.
2. (REST Contract) Edit Homepage News: https://github.com/DSpace/Rest7Contract/pull/45
   a. Delayed. General agreement (in meeting on March 21, 2019) that storing HTML in metadata fields is not really ideal behavior. Metadata (from a librarian standpoint) tends to be free of format-related markup (as that allows for easier sharing, understanding of metadata. Currently Community & Collection homepage information is HTML-based and *is stored in metadata* that is appropriate for a minor subset of information (like the title) but it is better to move large/rich text to bitstreams.
   b. Proposal here is to consider storing HTML-based markup (for Site, Community & Collection homepages) in Bitstream(s) associated with the object in question. May allow for more CMS-lite behavior in the future
   c. Timeline for this is uncertain. Possibly in 7 or 8. May depend on how/whether it can be scoped.

Notes