Requirements for Ingestion2 Admin Dashboard - Minimum Viable Product

The Admin Dashboard enables a user to define and execute harvests, mappings, enrichments, and indexing of DPLA provider metadata via a web interface.

Partner profile

- Create a partner profile that includes the official name of the organization.
- Add OAI feed to the partner profile, including the feed URL, the metadata prefix, whether sets are enabled and which sets to harvest.
- Add API to partner profile, including URL and parameters.
- Add static data download (ie. XML file) to the partner profile, including location.
- Add primary and secondary points of contact to the partner profile, including contact details.
- Edit/delete partner profiles.

Harvest jobs

- See all past harvest jobs associated with a provider, including when they started, when they stopped, how long they took and how many records were harvested.
- Start a harvest job.
- While a harvest job is running, receive feedback about its status, including when it started, how long it has been running, how many records have been harvested, and how many records are left to harvest.
- When a harvest job completes, receive feedback including how long it took, and how many records were downloaded.
- Stop a harvest job.
- When a harvest job is manually stopped, choose to either keep or discard those records which were harvested prior to stopping.
- When a harvest job fails, receive feedback including when it failed, the record number at which it failed, and information about why it failed.
- When a harvest job fails, choose to either keep or discard those records which were harvested prior to failing.
- When a single record within a harvest job fails (but the harvest job continues), receive feedback including the number of the failed record and information about why it failed.
- Delete all harvested records for a provider.

Mapping documents

- See all mapping docs associated with a provider, including current and past mappings, the dates they were uploaded, and any mapping jobs in which they were used.
- Generally one mapping will be the primary mapping, i.e. the one that is current, most likely the most recent. However, previous mappings should be stored and available for reference and historical purposes.

- Upload a new mapping as a ruby doc and associate it with a provider. Make it the primary mapping
- Download an existing mapping doc for edit or review outside of the system.
- Delete a mapping doc (only if it hasn't been used in a mapping job).

Mapping jobs

- See all mapping jobs associated with a provider, including when they started, when they stopped, how long they took, how many records were mapped, and which mapping docs were used.
- Define a new mapping job, including which mapping doc should be used, and which harvested records should be mapped (ie. all records, or 10% of records selected at random).
- Start a mapping job.
- While a mapping job is running, receive feedback about its status, including when it started, how long it has been running, how many records have been mapped, and how many records are left to map. This can be through a reload or through whatever way is efficient.
- Stop a mapping job.
- When a mapping job is manually stopped, choose to either keep or discard those records which were mapped prior to stopping.
- When a mapping job fails, receive feedback including when it failed, the record number at which it failed, and information about why it failed.
- When a mapping job fails, choose to either keep or discard those records which were mapped prior to stopping.
- When a single record within a mapping jobs fails (but the mapping job continues), receive feedback including the number of the failed record and information about why it failed.
- Delete all mapped records for a provider.

Enrichment Profiles

- Enrichment profiles are JSON documents that allow users to define and execute enrichment tasks. They include instructions to call a set of standard enrichments that come with the application. Enrichments might include DCMI normalization or punctuation enrichment.
- A “generic enrichment profile” which includes the most typical enrichments is preloaded in the system
- A user can edit that profile to customize it per provider.
- If the enrichment profile is edited, the previous version should be stored for future review or record keeping purposes.

Enrichment jobs

- See all enrichment jobs associated with a provider, including when they started, when they stopped, how long they took, how many records were enriched, and which enrichment profiles were used.
- Define a new enrichment job, including which profile should be used, and which mapped records should be enriched (ie. all records, or 10% of records selected at random).
- Start an enrichment job.
- While an enrichment job is running, receive feedback about its status, including when it started, how long it has been running, how many records have been enriched, and how many records are left to enrich.
- Stop an enrichment job.
- When an enrichment job is manually stopped, choose to either keep or discard those records which were enriched prior to stopping.
- When an enrichment job fails, receive feedback including when it failed, the record number at which it failed, and information about why it failed.
- When an enrichment job fails, choose to either keep or discard those records which were enriched prior to stopping.
- When a single record within an enrichment jobs fails (but the enrichment job continues), receive feedback including the number of the failed record and information about why it failed.
- Delete all enriched records for a provider.

Indexing jobs

- See all indexing jobs associated with a provider, including when they started, when they stopped, how long they took, how many records were indexed, and which server they were indexed to (staging or production).
- Define a new indexing job, including which server to index to, and which enriched records should be enriched (ie. all records, or 10% of records selected at random).
- Start an indexing job.

- While an indexing job is running, receive feedback about its status, including when it started, how long it has been running, how many records have been indexed, and how many records are left to index.
- Stop an indexing job.
- When an indexing job is manually stopped, choose to either keep or discard those records which were enriched prior to stopping.
- When an indexing job fails, receive feedback including when it failed, the identifier of the record at which failed, and information about why it failed.
- When an indexing job fails, choose to either keep or discard those records which were indexed prior to stopping.
- When a single record within an indexing jobs fails (but the indexing job continues), receive feedback including the identifier of the failed record and information about why it failed.
- Delete all indexed records for a provider.

Quality Control

- Currently the Blacklight-based QA interface can:
    - Provide records in a searchable interface
    - Facet in search results for
        - Type
        - Language
        - Subject
        - Format
        - Collection
        - Data provider
        - Creator
        - Place
    - Show mapped or enriched records side by side with the original record
    - Run validations for the presence of dataProvider, rights, title, isShownAt, preview, and type
    - Run reports on the unique values for most of the fields in the sourceResource class
- QA can be done after the mapping and enrichment stages if needed
- A few minor changes are needed to the current QA interface for the minimum viable product
    - see “blank” as a facet for no existing value
    - see a facet for pref and providedLabel for the properties currently faceted
    - field report for every property, both pref and providedLabel
    - validation reports in same CSV download format as field value reports
    - a thumbnail grid view
    - page through results from the side-by-side view
    - remove the “dashboard” link
    - When values are combined in the search result, combine with a “;” not a “,”

        - I.e. “Creator: McNatt, J. Dobbin; Benditt, Lauren”
            - not “Creator: McNatt, J. Dobbin, Benditt, Lauren”

Additional requirements

- See all jobs currently running and their statuses.
- See a complete list of all jobs that have run / are running for a given provider.
- Schedule a job to begin automatically at some point in the future.
- Require user login to access the application.

DRAFT Requirements for Future Developments

Partner Profiles

- Add more than one feed/data source to a partner profile. Potentially do this is some sort of multi-/bulk add fashion (i.e. do not have to reopen a form every time you need to add a feed)

Harvest

- Future need: harvest from more than one source as a single ingest

Pre-Mapping Analysis of data

This area describes work that is done on records from a data source *before* it is mapped. This analysis aids in the creation of the crosswalk/mapping.

- Download the data set as a single file for use in other statistical/analysis tools
    - XML from OAI feed
    - JSON from an API
- See a list of all properties used in the data set
- Get counts of the number of records containing a property (i.e. there are 32,000 records with <dc:title>)
- Get reports on the numbers of records missing properties (i.e. 5,000 records do not have <dc:title>)
    - it would be ideal to know this for any given property, but especially crucial for the DPLA required elements
- Get a list of all the unique values for a given property, with a count of how many times it exists
- See the total number of records
- Run a validation test with a pre-determined rubric for “completeness”

Mapping

- Run mappings written to metadata profiles other than the DPLA MAP and in formats other than RDF
- Run more than one mapping against a data set or ingest
- Share a mapping across more than one provider

Enrichments

- Create an enrichment activity and add it to the system code
- Configure an existing enrichment, such as set up a reconciliation activity for a property against a new LOD endpoint

Publishing/Exporting jobs

- Export data as RDF XML or JSON
- Be able to configure which metadata properties to expose or suppress from export/publishing
- Expose data as feed according to OAI PMH protocol
- Expose data according to ResourceSync specification