

MetacatUI - Story #7040

A client side data and metadata manager and editor

07/06/2016 02:32 PM - Chris Jones

Status:	In Progress	Start date:	07/06/2016
Priority:	Normal	Due date:	
Assignee:	Chris Jones	% Done:	35%
Category:		Estimated time:	0.00 hour
Target version:	2.0.0	Spent time:	0.00 hour
Description <p>We want to be able to provide basic metadata management and data upload for researchers using a client-side app that is a replacement for the Metacat Data Registry (server-side Perl script.) This ticket is the container story for this work. We need to develop requirements, mockups, and use cases to convey the scope of the work before (and possibly during) development. Since this is a major change in both functionality and UI layout, we'll target this for a 2.0.0 release.</p> <p>For development, we'll be extending the existing Backbone application with new classes that:</p> <ol style="list-style-type: none">1) Support the creation and parsing of DataONE SystemMetadata,2) Support the creation and parsing of OAI-ORE resource maps3) Support the creation and parsing of EML science metadata documents4) Support the future creation and parsing of ISO 19139 science metadata documents5) Support the future creation and parsing of FGDC CSDGM science metadata documents <p>Design documents are being developed here: https://github.com/NCEAS/metacatui/blob/master/metacatui/docs/design/editor</p>			
Subtasks:			
Task # 7041: Develop a design document to steer software development.			Resolved
Task # 7042: Develop mockup images for editor use cases			Resolved
Task # 7043: Develop use case documents for basic editor functions			Resolved
Task # 7044: Develop the 'Upload Data' use case			Closed
Task # 7045: Develop the 'Rename a file' use case			Closed
Task # 7046: Develop the 'Remove a file' use case			Closed
Task # 7047: Develop the 'Add Metadata' use case			Closed
Task # 7048: Develop the 'Update Metadata' use case			Resolved
Task # 7049: Develop the 'View Public Metadata' use case			Resolved
Task # 7050: Develop the 'Add a Nested Dataset' use case			Resolved
Task # 7056: Investigate schema-based JS object generation			Closed
Story # 7106: Retrieve an EML doc and parse into model attributes			New
Feature # 7107: Basic fields on the EML211 model			New
Feature # 7108: EMLParty modules			New
Feature # 7109: EMLKeyword modules			New
Feature # 7110: EMLOnlineDistribution			New
Feature # 7111: GeographicCoverage			New
Feature # 7112: TemporalCoverage			New
Feature # 7113: Taxon			New
Feature # 7114: EMLMethods			New
Feature # 7115: EMLProject			New
Story # 7116: Serialize an EML model to an EML document			New
Story # 7117: Serialize a System Metadata doc from a DataONEObject model			In Progress
Story # 7118: Create a new EML and its System Meta doc on a member node			New
Story # 7119: Render an EML model in the UI			New
Story # 7120: Overview section			New

Story # 7121: People section	New
Story # 7122: Dates section	New
Story # 7123: Locations section	New
Story # 7124: Taxa section	New
Story # 7125: Methods section	New
Story # 7126: Projects section	New
Story # 7127: Allow editing of EML display	New
Story # 7128: Make all fields in the EML View editable	New
Story # 7129: Update EML module models when their view fields are edited	New
Story # 7130: Update an EML doc and System Metadata doc on a member node	New
Story # 7131: Allow users to edit or create data packages in the editor	New
Feature # 7132: Serialize a Resource Map from a DataPackage collection	In Progress
Feature # 7133: Rename a data object by updating its System Metadata (via DataONEObject...	Resolved
Story # 7134: Remove a data object from a collection	New
Story # 7135: Show data upload form to the user in the DataPackage view	New
Story # 7136: Add a new data file to a package	New
Story # 7137: Create a data package from scratch	New

History

#1 - 09/30/2016 02:05 PM - Lauren Walker

- Subject changed from A client side data and metadata manager and editor needs to be developed to A client side data and metadata manager and editor
- Status changed from New to In Progress