

collecting_society - Konzept #340

GUI to create foreign objects

04/16/2017 05:14 AM - Alexander Blum

Status:	Erledigt	Start date:	
Priority:	Hoch	Due date:	
Assignee:	Alexander Blum	% Done:	100%
Category:		Estimated time:	1.00 hour
Target version:	Repertoire 1) Testing phase I	Spent time:	0.00 hour

Description

Sidenote for Meik: I renamed "2nd level objects" to "foreign objects" to emphasize the relation issues.

Context

We have two object creation processes, which need to enable the user to create objects of others, see [Specification](#). Especially for the creation of Creations the corresponding GUI/Form is quite complex:

- Our database has to be queried
- Future feature: additional databases should be queried (e.g. musicbrainz)
- Additional information should be provided for disambiguation (e.g. artists with the same name)
- If no match was found, a new object should be specified (artist: name, email; creation: name, artist)
- Unclaimed objects created by a web user should be editable afterwards

Options

Right now, I see four options for the implementation:

1. One Form + No Database requests (~3h)

We skip the database request part.

- The form is easy to implement with standard forms
- We might get lots of duplicates to handle later on
- The invitation of other artists would not be reasonable

2. Several Forms (+ No Database requests) (~10/30h)

We split the specification of relationships to other objects into several forms.

- Group members can only be added after the Artist was created (e.g. an "add" icon on the artist details page)
- Contributors and originals/derivatives can only be added after the Creation was created (e.g. an "add" icon on the creation details page)
- This would break down the complexity
- This would be more error-prone as the user is not strictly guided anymore (could be countered e.g. by notes or big icons on the object details page).
- The form for database queries would still need to be hand crafted, although we have more degrees of freedom (e.g. to change the whole form, if there's no match).
- We could omit the database queries in the beginning (like in option 1) to save time (~20h)

3. One Form + Wizard (~40h)

We defer the creation of foreign objects and divide them into steps.

- The form class of portal would enable us to implement something like this.
- This could break down one complexity but add another
- There should be feedback for the steps (e.g. 3 artists and 2 creations left)
- The association should be clear (this artist is the contributor for lyrics, etc.)
- We might be able to reuse parts of the usual forms
- We could offer the whole form, as the user might be willing to provide more information

- I would not be convinced to call this a nice gui
- The implementation might need similiar time as the 4th option

4. One Form + Layover (~50h)

We provide a complete hand crafted form.

- The main object creation form
 - A dropdown list for database queries with autotype selection
 - More information on mouseover (ajax calls)
 - An "add" icon in the dropdown, if no match was found
- A layover with the form for foreign objects on "add"
 - Creation of the foreign object after submission (ajax call)
 - Refresh of the dropdown list and autoselect the new object (ajax call)

Opinion

On one hand, we have limited time for the implementation on the other hand, the creation form is the core of the metadata part of repertoire.

I would cross out option 3.

If we have time, I'd prefere option 4.

Otherwise I would vote for option 2 without database queries and add those, if we have time left. Those could be developed into option 4 (with ajax calls) later on.

Opinions? Other proposals?

Related issues:

Related to collecting_society - Konzept #324: Should objects of others be edi...

Erledigt

Related to collecting_society - Webfrontend #387: Complete creation views

Erledigt

History

#1 - 04/16/2017 06:17 AM - Alexander Blum

- Related to Konzept #324: Should objects of others be editable, if you have created it? added

#2 - 05/01/2017 02:55 PM - Meik Michalke

- Assignee changed from Meik Michalke to Alexander Blum

I would vote for option 2 without database queries and add those, if we have time left. Those could be developed into option 4 (with ajax calls) later on.

yes, starting with a basic 2 and then moving towards 4 over time sound like the way to go for me: gets us to a generally usable result rather quick, but also includes a plan of improvement for the future.

#3 - 05/27/2017 07:01 PM - Alexander Blum

- Related to Webfrontend #387: Complete creation views added

#4 - 05/27/2017 07:04 PM - Alexander Blum

- Status changed from Feedback to Erledigt

- % Done changed from 0 to 100

starting with a basic 2 and then moving towards 4 over time sound like the way to go for me

allright.

#5 - 10/08/2019 03:34 PM - Alexander Blum

- Target version changed from 1) Testing phase I to Repertoire 1) Testing phase I

#6 - 10/08/2019 03:50 PM - Alexander Blum

- Project changed from repertoire to collecting_society