Shakespeare™  
Your Metadata Management System

# Introduction

The aim of this document is to identify all tasks to be fulfilled utilizing the Shakespeare application. Tasks are linked to workflows and tasks change the content of the applications data store.

# The Story Behind

…

# User Stories

## Introduction

The User Stories section contains all demanded interactions with the Shakespeare application. An intended by agile manifestos there are written down in a pragmatic and simple manner, in short form. They follow this syntax: As a <role> I want to <goal> so that <benefit>. Each story is labeled by a unique heading. They are grouped by the role.

## User

### Connect

As a user I want to connect to the application by simply navigating to the URL to start my metadata research work.

### Search

As a user I want to search the content of the application in a Google like manner and get the accessible results represented in a structured way. I need to do so to access the information I’m looking for.

### Switch language

As a user I want to change the display language

## Guest User

### Login

As a guest user I want to be able to login into the application to maintain metadata. After logging in I would like to act according to my associated profile(s).

## Named User

### Logout

As a named user I want to be able to logout from the application. After logging out I expect to act as a guest user only.

## Translator

### Translate

As a translator I want to translate existing poems into any available target language to make the content accessible to other information consumers.

# Workflows

The orchestration of tasks produces workflows. Thus the following sections contain all the tasks, which are described in a common way: 1st is the short definition of what to expect to be done within this task and 2nd the possible interactions, which lead to other tasks. A leading “a:” indicates that an interaction depends on authorizations. Possible interactions are described in tabular form containing the following columns.

|  |  |  |  |
| --- | --- | --- | --- |
| Possible Interaction | Interaction element[[1]](#footnote-1) | Content[[2]](#footnote-2) | Target Task |

## Basics (01)

Basic tasks are common tasks which could be performed from (nearly) everywhere in the application. Included are login and logout as well as search. Some mixture is with languages. A basic task is to change UI language, more specific tasks will handle translation etc. for poems. They build their own section (see Internationalization (03)).

### Display Home 01.01

Display authorized menus, navigation, content and status information (like logged in user etc.). All things represented depend on the authorization of the logged in user or the guest user. Initially the UI language is picked from the browser. The representation of menus and other interaction elements reflect the current UI language, the navigation and content may vary in case of the selection of a Poem language.

|  |  |  |  |
| --- | --- | --- | --- |
| Manual Login / Logout[[3]](#footnote-3) | Dedicated Button | Content remains unchanged; except there is no longer a valid authorization, then content is cleared | Manual Login 01.02/ Logout 01.03 |
| Select Poem | Link | Latest visible revision for the selected poem in current languages | Select Poem 02.01 |
| Select UI Language | Selection list | Latest visible revision for the selected poem in new UI language | Select UI Language 01.04 |
| Select Poem language | Selection list | Latest visible revision for the selected poem in new UI language |  |
| Search | Text box + Icon | Search result | Search 01.05 |
| a: Create Poem | Menu Item[[4]](#footnote-4) | Content moves from current selection to edit poems core features to edit content of first revision | Create Poem 02.03 |
| a: Admin | Menu | Admin screen | 04.01 |
| Display Home | Menu | Clear content | Display Home 01.01 |

After navigating to the home page any selection is reset to NIL and the initial navigation is represented. The content pane is empty except eventually existing notes and the welcome screen (see content pane chapter for ideas).

### Manual Login 01.02

Provide logon screen and do the logon. The logon process could be interrupted. In case of logon the user id is stored for the session and reused for all authorizations. Log in or cancelation both leads to a return to the Home screen.

|  |  |  |  |
| --- | --- | --- | --- |
| Submit | Dedicated button | Content remains unchanged | Home 01.01 |
| Cancel | Dedicated button | Content remains unchanged | Home 01.01 |

### Logout 01.03

This task does not contain any interaction. By pressing the button on home screen the currently logged in user is logged out and the view returns to the home screen. The stored user id is removed from the session.

### Select UI Language 01.04

By selecting an alternate UI language on home screen a pop-up dialog is presented asking a logged in user to store the selected language permanently in the user profile. After that or for guest users the view returns to the home screen.

### Search 01.05

This Google like search scans all properties of all Poems and displays the result grouped by poem type.

## Poem related (02)

Poem is the central entity of the application even it does not contain the information. It is used to navigate, but never displayed. And it is used to structure your information. Most of the actions taken on a poem lead to action performed with other objects like Revision or Relations. The following sections describe possible tasks related with poems.

### Select Poem 02.01

From the main navigation or from search results a single poem could be selected. This leads to two different tasks. Internally the selection is stored, so the navigation is adjusted on the next refresh (beside the selected poem it contains the predecessors and successors if available). Secondly the content is switched to the latest accessible revision of the newly selected poem. There is no interaction at this task.

As poems will not be displayed, but the latest accessible revision, this task only performs the selection of this latest accessible revision according to the user authorizations and the redirection to show this revision. If there are only inaccessible revisions the user is informed about this by redirecting to the home page and displaying a note there. If there are no revisions and the user is allowed to create one he is redirected to edit the first revision. If there are no revisions and the user is not allowed to he is redirected to the home page and informed about this by displaying a note there.

|  |  |  |  |
| --- | --- | --- | --- |
|  |  | Latest accessible revision exists? | |
|  |  | Yes | No |
| User allowed creating revision? | Yes | Redirect to display latest accessible revision. | Redirect to editing the first revision |
| No | Redirect to home page and informing about this situation |

### Create Poem 02.03

Creating a poem contains two tasks: 1) Create the poem itself and 2) create the first revision of it. After creating both objects the content is switched to the revision to be edited. Process could be interrupted at any stage.

### Edit Poem 02.04

Draft: Poems are not to be edited. Changing ones poem type is prohibited, changing the original language is difficult. Editing poems shall not be offered with the application. Manually entering a path to edit a poem shall lead to a redirection to the home page with an informative note or a redirection to an editable revision. Decide (see to do section)!

### Delete Poem 02.05

Idea: Poems could not be deleted but switched into an inactive mode.

### Translate Poem 02.06

Draft: Poems are not translated but their revisions. The poem does only contain the original language so that it is always clear what the core revisions and what translations are.

## Internationalization (03)

## Administration (04)

## Revision (05)

Revisions are the hidden objects behind. They are addressed during the display of information as well for editing and translating. Also revisions could be part of a lecture process (approval, denial of changes, etc.)

### Select Revision 05.01

Idea: Offer a list of all existing revisions as a list and allow selecting from it.

### Show Revision 05.02

As poems are not displayed, but the latest accessible revisions, this task is performed whenever a poem was selected (Select Poem 02.01). When reaching this stage the latest accessible revision is displayed. Based on this various interactions are possible.

|  |  |  |  |
| --- | --- | --- | --- |
| Edit | Dedicated button | Content is switched from displaying this revision to creating a new one | Create Revision 05.03 |
| Delete | Dedicated button | Clear content | Home 01.01 |
| Translate | Selection list | Content is switched from displaying a poem into translating it; languages are the original language (not poem display language) and target language as selected from list |  |

### Create Revision 05.03

### Edit Revision 05.04

Draft: Revisions are not edited. The action to be performed mainly depends on the language of the revision compared to the original language of the poem.

### Delete Revision 05.05

### Translate Revisions 05.06

# Content

This chapter outlines the various content panes. The content pane is between the main menu and the bottom status line and left from the main navigation.

|  |  |
| --- | --- |
| Main menu | |
| Main navigation | Content pane |
| Status | |

## Home

Idea: Display search functionality in the centre of home content pane. Like google.com. The search box and some links to vendor homepage, a impressum or support information.

## Poem

## Revision

## Relation

## Search Result

## Admin

# To do’s

* Check the name of the server; it is not Shakespear but **Shakespeare**!
* Add column status add poem level. As things shall not be deleted the status shall indicate whether a poem is active (and thus selectable and editable) or inactive. Maybe it is better to call the column ACTIVE instead of STATUS to just reflect this information as Boolean information.
* Decide how to handle invalid paths (like edit poem): Redirect to home page with displaying a note or redirect to possible valid interaction (like edit revision).
* Switch

# Table of Content

Introduction 1

The Story Behind 1

User Stories 1

Introduction 1

User 2

Connect 2

Search 2

Guest User 2

Login 2

Named User 2

Logout 2

Workflows 3

Basics (01) 3

Display Home 01.01 3

Manual Login 01.02 4

Logout 01.03 5

Select UI Language 01.04 5

Search 01.05 5

Poem related (02) 5

Select Poem 02.01 5

Create Poem 02.03 6

Edit Poem 02.04 6

Delete Poem 02.05 6

Translate Poem 02.06 6

Internationalization (03) 7

Administration (04) 7

Revision (05) 7

Select Revision 05.01 7

Show Revision 05.02 7

Create Revision 05.03 8

Edit Revision 05.04 8

Delete Revision 05.05 8

Translate Revisions 05.06 8

Content 8

Home 9

Poem 10

Revision 10

Relation 10

Search Result 10

Admin 10

To do’s 10

Table of Content 10

1. It’s important to mention the interaction element here. As nearly all interactions in web pages are triggered by clicking links the interaction element contains a standardized description of the visual representation like Link, Dedicated Button, or Menu Item etc. [↑](#footnote-ref-1)
2. Content describes what is displayed beside the main menu and the navigation at the centre content pane of the application after the process finished, which was triggered by interacting with the application. [↑](#footnote-ref-2)
3. Manual login is only presented in case no user is logged in. Logout is only presented in case a user is logged in. [↑](#footnote-ref-3)
4. Menu Item implies that there is a Menu holding this item. Menu shall be labelled “Poem”, selecting it shall lead to nothing. [↑](#footnote-ref-4)