# Changelog yuuvis RAD 9.4

| Schlüssel | Zusammenfassung | Beschreibung |
|---|---|---|
| ERA-8445 | Improve UI of properties view for deadlines and delays | As a modeler I would like to have better overview of deadline configuration options in order to better understand the viable combinations and when a configuration applies.<br><br>Acceptance criteria:<br><br>• **Deadline / Delay behavior** (Configure the behavior when start point is reached again)<br>  ○ Provide three comboboxes, that cover three situations/behaviors when starting point is reached<br>  ○ When already fired / passed<br>    ▪ nothing<br>    ▪ create new deadline (with same parameters as this one) (default)<br>  ○ When already stopped / suspended<br>    ▪ nothing<br>    ▪ create new deadline (default)<br>  ○ When still ticking<br>    ▪ nothing<br>    ▪ Extend existing time period (default)<br>    ▪ create new deadline<br>    ▪ Existing models that have checked both options are updated so that extend option applies (to be communicated in digest)<br>  ○ The tooltips for these options shall be updated / clearer if necessary (currently they all look quite the same, which is confusing)<br>• **Action options** define actions when a deadline just passed (fires)<br>  ○ E-Mail notifications<br>  ○ Work item actions<br>• The tooltips shall be doublechecked (we say "deadline is triggered" and "deadline is expired" which may confuse user: two words for the same thing - deadline expired and fired events configured to be activated on its expiry)<br>• Rename Verlauf/History to Auslösepunkte/Trigger points |
| ERA-9203 | Documents that are used in a process can be deleted only with an additional parameter | As a user, I would like to be prevented from accidental deleting of a document that is part of a process, in order to preserve the correctness of the process.<br><br>Acceptance criteria:<br><br>• Document-Delete operation checks whether document is being used in a process, and if it is, the delete operation fails<br>  ○ The documents that are in a folder that is about to be deleted shall be checked, and if one of them cannot be deleted due to its use in a process file, then the folder cannot be deleted as well<br>• Document-Delete operation gets a new parameter "forcedelete" that allows user to delete a document, even if document is being used in a process |
| ERA-9360 | Inbox entries are only marked as "substitute" if the user does not see it directly | As a user of the inbox I want entries in my inbox marked as "substitute" if and only if I would not see the entry without the substitution. This makes it more clear to me which entries are mine and which belong to a substitution<br><br>Acceptance criteria:<br><br>• As a user of the inbox I want entries in my inbox marked as "substitute" if and only if I would not see the entry without the substitution |
| TUK-1775 | The file encryption can be used for external archive storage as well | As an administrator, I want to be able to configure the file encryption for an external archive storage as well so that I can match the policies of my organisation.<br><br>Acceptance criteria:<br><br>• In the management studio, the 'Key' for the encryption can be selected for an external archive storage.<br>  ○ The label 'Key'/'Schlüssel' is changed to 'Encryption key'/'Encryption-Schlüssel'.<br>• The core-service is encrypting the files as configured for the external archive storage as well. |
| TUK-1853 | Programmers can request the new Signing-Service to send the document file for the given objectId to DocuSign | As a programmer, I want to be supported by an endpoint of the signature service that supports sending the document file for the given objectId to DocuSign so that the specified signers can sign it and that the object status is updated and an audit entry reported.<br><br>Acceptance criteria: |

- The Signing-Service of the consulting team is taken over by the development team and can be set up into the service management infrastructure.
  - The service is listed in the service admin as Signing-Service.
- The code is refactored:
  - Instead of the internal database, all necessary data is held in the object including the DocuSign eveleopeId and the status.
  - The labels of the service are generalized: no 'docusign' is shown in the Swagger-UI or in the endpoint.
- The service offers the endpoint .../signing/request/{documentId} to request the signing. In the Swagger-UI it is possible to request the signing with the following optional body: All properties are configured with an default value in the configuration file sining.yml:

- - signingPlatform - specifies the signing platform to be used. Currently only 'docusign' is possible
  - emailSubject - specifies the email subject to be used for the email that is sent by DocuSign
  - emailBody - specifies the email body to be used for the email that is sent by DocuSign
  - templateId - will come later: specifies the templateId that is configured on the DocuSign platform that defines the way to sign. This is optional.
- The singerOrder controls the sequence of the signers. If no order is given all signers are getting the email in parallel. Otherwise, a signer gets the email after his successor has signed.
- If no signers are given in the request body the signers data will be read from the object itself.

```
{
  "signingPlatform": "docusign",     / optional, default set
in authentication-prod.yml
  "emailSubject": "About why signature is requested for this
document.",     / optional, default set in authentication-prod.
yml
  "emailBody": "Detailed about why signature is requested for
this document.",    / optional, default set in authentication-
prod.yml
  "signers": {   /mandatory if no signer data are saved for
the given object
    [
      {      / one signer is mandatory
        "signerEmail": "paul.mustermann@myemail.com",
/mandatory
        "signerName": "Paul Mustermann",
        "signerOrder": 1    / optional
      },
      {
        "signerEmail": "emely.mustermann@myemail.com",
/mandatory
        "signerName": "Paul Mustermann",
        "signerOrder": 2
      },
    ]
  },
  "options": {            / optional
    "templateId": "3ed7cd27-a425-4fbe-b402-01ea9b5a7c2b",
    "signatureType": "qualified"  (tbd: check)
  }
}
```

- The request responses with error localized messages if properties are missing. tbd in the next story: list error cases by code
  - The signing service is not correctly configured. Contact your administrator. The signing services logs:

- - - property 'signingPlatform' is not or not correctly configured
      - The property 'emailSubject' is not configured
      - The property 'emailBudy' is not configured
    - The signing platform responded with an error. Contact your administrator.
    - No signer data are given. At least one signer with his email address has to be requested.
    - The object could not be updated because the object type does not possess the necessary signing properties.
  - The initial status of the signing process as well as the signer data, emailSubject and emailBody is saved as metadata to the object.
    - A custom history (audit) comment is written: "Signatures requested"
    - In this case of an error, an error is reported in the error property of the object.
  - The Signing-Service requests the DocuSign status for all documents with the yuvsigstatus = 'inprocess' but logs the result only.
    - Next Story (TUK-1953): Update the objects if DocuSign has changed its status.
  - For the review: The service can be requested via custom action for an object that is based on the given quasi-standard abstract object type (see referenced story).

| | | |
|---|---|---|
| TUK-1941 | Users are supported by digital grouped data fields in processes | Acceptance criteria:<br><br>• digit grouping shall be added as a data field property, that is offered<br>  ○ on creation of integer, long, decimal data field<br>  ○ on creation of user-defined data type that is derived from decimal<br>• core-service sends data in correct format so that client can render it correctly |
| TUK-1953 | The signing service is polling as configured for the status of the files that are sent do DocuSign and transfers a signed document back | As a programmer, I want that the signing service that I have requested for signing a document file with DocuSign polls as configured for the status of the files and that it transfers the signed document back as a new version so that the users can handle it.<br><br>Acceptance criteria:<br><br>• The service is polling for the status using for each of the documents with the propperty yuvsigning.yussigstatut = 'inprocess'.<br>• In this case, the file is partially signed the signer status of the object is updated ( old yuvsignersstatus = 'signed', new yuvsignersstatus = 'inprocess')<br>• In this case, the file is completely signed it is fetched from DocuSign, and saved as a new version of the object including the status update for the last signer.<br>• In this case of an error, the object is updated with the error message.<br>• In this case, a signer has revised this status is updated as well.<br>• For all specific object or content updates a specific audit entry is written as a custom comment.<br>• An endpoint for refreshing the DocuSign status is available. |

| TUK-1963 | Users can request signatures for a focused object | As a user, I want to be able to specify which users should sign the document I have focused. |

Prerequisite: The new Signing-Service is available.

Acceptance criteria:

- The signatures can be requested by a custom action 'Request signatures'/'Unterschriften anfordern' for objects of type document and mimetype 'application/pdf'. The object action 'Request signing' is only shown if
  - the object type is derived from the quasi standard abstract object tpye 'yuvsigning' as described here: https://yuuvisdevelop.atlassian.net/wiki/spaces/onpremise/pages/426770437/The+signing+of+a+document
  - the user has edit rights
  - the object is not finalized
  - nothing was requested yet (yuvsigstatus is NULL)
- After pushing the action a dialog is opened that allows you to enter
  - yuvsigemailsubject with label 'Email subject'/'E-Mail-Betreff', optional, but if entered: min 3 and max 100 characters
  - yuvsigemailbody wht label 'Email message'/'E-Mail-Nachricht', optional, but if entered: min 3 and max 4000 (to be checked on DocuSign-Side) characters
  - the list of signers, min 1 and max 6 (default, configurable in main.json with '"maxSigners": 6' in section '"signers":{}' (same default as set for signing-service)
    - yuvsigners.yuvsignerorder with label 'Signing order'/'
    - yuvsigners.yuvsignername with label 'Name', min 3 and max 100 characters, autocomplete is possible
    - yuvsigners.yuvsigneremail with label 'Email address'/'E-Mail-Adresse', autocomplete is possible
    - yuvsigners.yuvsignerphone, but later because of more complexity: 'Phone number'/'Mobilnummer' for Moxis
    - yuvsigners.yuvsignexternal, but later because of more complexity: 'external'/'extern ' for Moxis
  - After pushing the button 'Request'/'Anfordern' the signer data are written to the object and the endpoint of the Signing-Service with the necessary objectId is called and the first signer gets the email from DocuSign.
  - After pushing the button 'Cancel'/'Abbrechen' the dialog is closed. If data were entered the user is informed about unsaved data and whether he wants to cancel, same for  pushing ESC and icon 'x'.
- The aspect 'Signatures' is shown if the abstract object 'yuvsigning' is part of the system schema.

| TUK-2060 | If an object is contained in a process file and is requested to be deleted by the user an information about this is given and the deletion can be forced | As a user, I would like to be prevented from accidentally deleting an object that is part of a process, in order to preserve the correctness of the process. |

Acceptance criteria:

- The document-delete operation checks whether the document is being used in a process, and if it is, the delete operation fails. The user is informed about this situation and he is asked whether to delete nevertheless:
  en: 'This object is contained in a process file. Do you want to delete it nevertheless?' -> 'Cancel' / 'Delete' buttons are offered
  de: 'Dieses Objekt ist in einer Prozessakte enthalten. Soll es dennoch gelöscht werden?' -> 'Abbrechen' / 'Löschen'
  - Dialogue title: 'Effect deletion'/'Löschung erwirken'
  - The button 'Delete' is focused and the user can push the 'Enter' button to delete the object
  - Pushing the 'tab' key toggles between the two buttons
  - The title of the object is shown above the information.
- The document-delete endpoint is called again but with a new parameter that allows deleting it although it is used in a process.
- The relevant client API documentation is extended: https://yuuvisdevelop.atlassian.net/wiki/spaces/onpremise/pages/319620180/Client+Plugin+API#ClientPluginAPI-scope.api.http.del

| TUK-2092 | The deletion operation allows to force the deletion if the object is contained in a process file | As an administrator, I want to be able to force the deletion operation to delete an object although it is contained in a process file if this is required by my organization. |

Acceptance criteria:

- The delete operation in the management studio is extended with the new parameter forcedelete. If forcedelete=false the objects that are contained in a process file are not deleted and this is protocolled.
  - EN: "Delete objects that are contained in a process file"
  - DE: "Dokumente, die in einer Prozessakte enthalten sind, löschen"

| TUK-2332 | Users can find documents without files | As a user, I will be able to find documents without files so that I can attach the missing files to them. |

Acceptance criteria:

- In the search aspect 'Document files' the section 'Files size' offers the new aggregation item 'Without file'/'Ohne Datei'.
- It is possible to select this search criterion and the hit list lists the correct objects.
- This criterion is used in the saved searches as well.

| TUK-2430 | The service manager setup is installing the Signing-Service as well | As an administrator, I want to be supported by a service setup that installs the Signing-Service so that I do not have to configure the service manually. |

Acceptance criteria:

- There is a GIT pipeline that creates the delivery artifact.
- The service manager setup installs the Signing Service as well.
- In this case of a new installation, all necessary configurations are added.
- In this case of an update installation, the necessary configurations have to be added manually.
  - Information supports the administrator on what to configure.

| TUK-2431 | Users are informed about the status of a signing process in the object details aspect 'Signatures' | As a user, I want to see all information about the signing process so that I know about it and can talk about it. |

Acceptance criteria:

- A new aspect 'Signature'/'Unterschriften' is offered.
  The aspect presents the property values of the abstract object type:
  - If signatures were requested (property: yuvsigning.yuvsigstatus = !NULL)  the following information is given
    - A header 'Digital signatures'/'Digitale Unterschriften'
    - 'Request date'/'Anforderungsdatum' <date and time>
    - 'Requested by'/'Angefordert von' <user name, firstname (login name)>
    - 'Subject'/'Betreff' - property: yuvsigning.yuvsigsubject
    - 'Status' - yuvsigning.yuvsigstatus, this is a catalog, and its values are localized
    - 'Error message'/'Fehlermeldung' - property: yuvsigning.yuvsigerror
    - 'Signers'/'Unterschriftgeber' with a list of signers with their properties:
      - 'Signing order'/'Unterschriftenfolge' - yuvsigning.yuvsignerorder
        This number is shown in a gray circle with a tooltip.
      - 'Name' - yuvsigning.yuvsignername
      - 'Email'/'E-mail' - yuvsigning.yuvsigneremail
      - 'Status' - yuvsigning.yuvsignerstatus, this is a catalog, and its values are localized and the following colors are shown for the status values:
        - Gray for requested and inprocess
        - Green for signed
        - Yellow for revised, canceled, expired
        - Red for error
      - 'Date'/'Datum' - yuvsigning.yuvsignerdate
      - 'Comment'/'Bemerkung' - yuvsigning.yuvsignercomment
- All these signing properties are not shown in the object summary

| TUK-2502 | Users are supported by autocomplete suggestions in table column types that can be configured | As a user, I want to be supported by autocomplete suggestions during editing a table field so that I don't have to type much and can reduce typos. |

Acceptance criteria:

- Autocomplete suggestions are offered during editing table rows. The following field types are supported but must be configured for autocomplete indexing in the designer: string, catalog.

| TUK-2531 | It is possible to select the full-text language package for Europe or Asia during the installation | As an administrator, I want to be able to select the full-text language package license for Europe if my company wants to use a language from this package or for Asia. |

Acceptance criteria:

- During the installation, the language package can be selected. 'America & Europe' is the default. If 'Asia' was selected its license file is installed.
- The documentation is informed to mention that 'Asia' is only to be used by users of the application Kyocera Enterprise Information Management.

Notice: The Asia license contains English, French, Traditional-Chinese, Hangul, Thai, Simplified-Chinese, Japanese, Indonesian, and Malaysian.

| | | |
|---|---|---|
| TUK-2572 | The files of Hancom Office 'word', 'cell', and 'show' files are classified as 'word', 'excel', and 'powerpoint', and files of format '7z' and 'rar' as 'zip' | As a user, I want to be able to search specifically for Hancom Office word, cell, and show files so that I get fewer hits. |

Background: KDC Korea wants to offer the support of Hancom Office documents via rendition service. But we are currently not supporting the mimetype of these files not. So every Hancom file gets the filetype octet-stream.

Acceptance criteria:

- Hancom 'word' files with mime-type application/haansoftdocx can be found with filter 'word'.
- Hancom 'cell' files with mime-type application/haansoftcell can be found with filter 'excel'.
- Hancom show files with mime-type application/haansoftshow can be found with filter 'powerpoint'.
- 7z files with mime-type application/x-7z-compression and rar files with mime-type application/x-rar-compression can be found with filter 'zip'.
- html files are classified as 'txt'

16 Vorgänge