

OBJECT : To be able to manage different versions of an entity which has the same uuid values.

OUR NOTES :

**Within this sprint we initially decided to start working with People entity.
So all below scenarios,notes and solutions should only be considered for People entity.**

Rule Set :

Rule 1 : Unapproved("DRAFT") entities should not be visible to other users other than it's creator and Admins / Super Users (This approach is mentioned on Epic 552 : Below Vocabulary title , see version description)

Rule 2 : Any entity should be immutable : If a user wants to change it , he should have a new service call like "Create New Version" and it should create a new entity/version on that table.No hard update/delete operation should be allowed.

Rule 3 (Not sure yet) : If a person entity does not have a published version yet then that person should not be allowed to assigned to any position (Supported or Vacant Positions) . This operation should not be allowed and should warn operator with a message for example : "This person entity does not have a published version yet" .

SCENARIOS :

Scenario 1

- An Advisor wants to create a New Report
- He can not find the person he is looking for
- He decides to create a new person (A new button should be placed anywhere on New Report page)
- New person entity is recorded into People table with the status 'Draft',with new uuid,hash value,ownerUuid,versionTime .. These will be new columns for Entities.
- An email is sent to admin that an Advisor created a new person waiting for his approval.
- This new person will only visible for the creator of that person and Admin/Super User because of it's status (Draft)

Scenario 2 :

- An Advisor wants to create a New Report
- He finds the person (Because this person was previously approved by Admin and it's status is in Published state)
- Person entity has missing/wrong info and Advisor wants to update it
- Opens the page of related person and clicks **Create New Version** button
- Updates the related data that he wants to change
- Updated person entity is inserted into People table as a new version with the status 'Draft',with the same uuid but different hash value and timestamp values ... , the previous version remains the same in table
- An email is sent to admin that an Advisor modified an existing published person which means a new version is created as Draft
- Admin opens related Person's page ,checks new version and if he thinks it's ok then he approves it as new published version.Previously published version status is set to Draft status in background right after.

Scenario 3 :

- An Advisor wants to create a New Report
- He finds the person (It's status is in Draft state but this person entity was created previously by this user himself so he is able to see this person entity)
- Person entity has missing/wrong info and Advisor wants to update it
- Opens the page of related person and clicks **Create New Version** button
- Updates the related data that he wants to change
- Updated person entity is inserted into People table as a new version with the status 'Draft',with the same uuid but different hash value and timestamp values ... , the previous version remains the same in table

- An email is sent to admin that an Advisor modified an existing person which also means a new version is created

Scenario 4 (Not sure for the solution yet):

- A user (let's name it Tom) searches for a person (Let's name it Eric) but he can not find Eric
- But assume that there is a person already named Eric , the problem is that this person was created by another Advisor (Let's name it by Martin). This person entity (Martin's Eric) will not be visible for Tom because Admin did not approve this entity yet (according to our rule set : Not-Approved and Not-Owned entity versions will not be visible to other users)
- So this means that Tom will want to create a new person entity and as a result this new Eric's uuid will not be the same with the previously created Eric's uuid .
In a situation like that we should know that these 2 seperately created users are the same indeed and we should handle this case and should not let Tom to create a new Person with similar data which will cause duplicate data.

May be a partial solution ?? : When Tom starts to enter the info of the person on create person page , the system will start to make an instant search in background with entered info (May be not for all info ,for instance may be for E-Mail,Phone Number and Name) and if system finds a similar user in the system , it just shows a warning message to Tom like "There may be already a person in ANET , please contact with your admin if you can not see that person on search results" (system will not show any detail info about that user and will not let Tom to select that user because it is not approved yet and also it was not created by Tom)

However, in this case, the Advisor will have to wait for the Admin's approval, which is actually one of the situations we want to get rid of.

Scenario 5 :

- A Person entity with name John has 4 versions , all these 4 versions are in Draft state
- Advisor wants to search a Person (**creator of this Person is Advisor himself**)

Which version should be presented :

The version that has most recent timestamp value should be presented in search results

In detail page of that person he should also be able to see links (because he is the operator that created this person previously) to other draft versions.When he clicks one of that link , another info page related with that version should be presented.

Scenario 6 :

- A Person entity with name John has 4 versions , one is published and the other 3 versions are in Draft state
- Advisor wants to search a Person (**creator of this Person is Advisor himself**)

Which version should be presented :

- The published version should be presented in search results

- In detail page of that person he should also be able to see links (because he is the operator that created this person previously) to other draft versions.When he clicks one of that link then another info page related with that version should be presented.

Scenario 7 :

- A Person entity with name John has 4 versions , all these 4 versions are in Draft state
- Advisor wants to search a Person that has name John (but creator of this Person is another Advisor)

Which version should be presented :

- None of person version will be displayed in search list because this entity does not have a published version and also this entity was not created by this Advisor – So we will be looking into another scenario (**Please see Scenario 4**)

Scenario 8 :

- A Person entity with name John has 4 versions , one is published and the other 3 versions are in Draft state
- Advisor wants to search a Person (creator of this Person is another Advisor)

Which version should be presented :

- Only the published version should be presented in search results
- In detail page of that person he should not be able to see links to other draft versions because this Person is not created by him