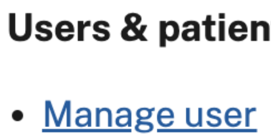
Manage users (Support admin)

The manage user feature will be part of the users & patients section in the support admin landing page.

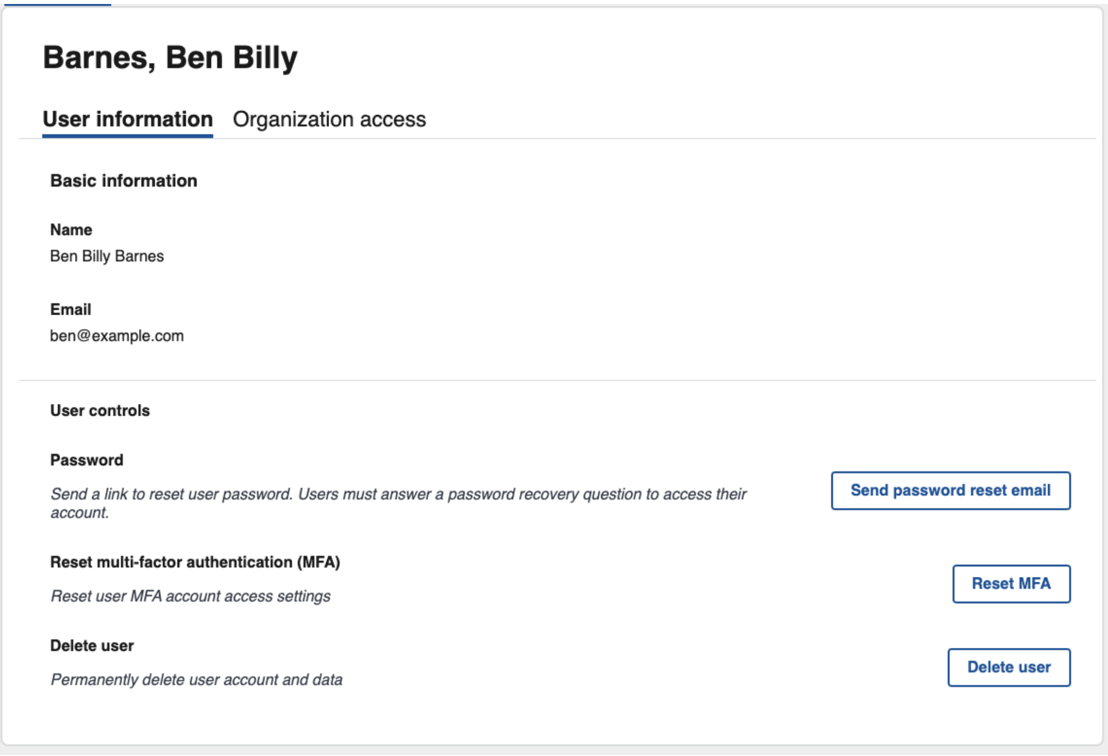


Users & patients

- Manage user

User view (Manage user)

This page will serve as the sole page to check and trigger updates to a user account. The page will be linked directly from the support admin menu page (Manager user link). The page will display a search bar at the top that will allow the support admin to search by email. The search bar will have a search button and will validate that the content entered is a valid email address (regex pattern). Upon searching and finding a match the user's account will load the user view component that displays all the elements shown in the images below.



Barnes, Ben Billy

Basic information

Name
Ben Billy Barnes

Email
ben@example.com

User controls

Password
Send a link to reset user password. Users must answer a password recovery question to access their account.

Reset multi-factor authentication (MFA)
Reset user MFA account access settings

Delete user
Permanently delete user account and data

Notes

- Using the `setUserIsDeleted` does not remove a user from any groups – undeleting them restores them back to their previous groups

Query to populate this page (Elisa will check) and make sure we have all the necessary data to render the actions and organization access. Elisa Lee

- User settings page currently uses `getUser` method to load user information
- Data that is returned is:

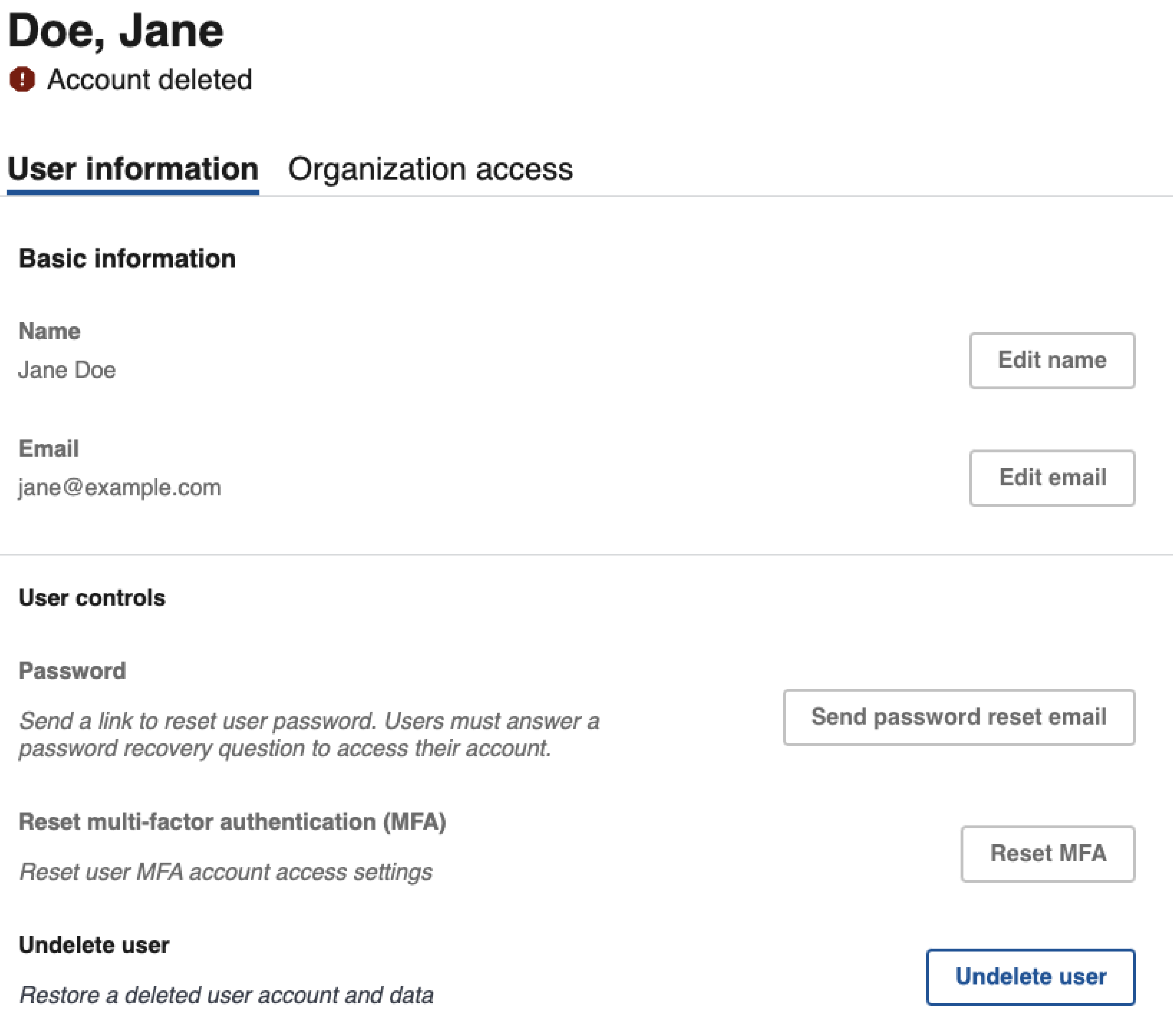
```
{
   "data": {
      "user": {
         "id": "4b1af6cf-b696-40ba-a826-ec9988e762ba",
         "firstName": "Test deleted facility",
         "middleName": null,
         "lastName": "User last",
         "roleDescription": "Test-entry user",
         "role": "ENTRY_ONLY",
         "permissions": [
            "UPDATE_TEST",
            "SEARCH_PATIENTS",
            "START_TEST",
            "SUBMIT_TEST"
         ],
         "email": "elisa+testdeletedfacility_dev5@skylight.digital",
         "status": "ACTIVE",
         "organization": {
            "testingFacility": [],
            "__typename": "Organization"
         },
         "__typename": "User"
      }
   }
}
```

- However, `getUser` method uses `findById` method which will not return soft deleted users
- We may want to create a new getUser method that uses `findByIdIncludeArchived` method to ensure soft deleted users are also returned in this view

Suggestions

If an account is in the deleted state the application should not allow for any other updates to it besides undeleting it. We can follow the behavior of the deactivated state, which disables all controls in the user view page.

Undelete user



Doe, Jane

Account deleted

Basic information

Name
Jane Doe

Email
jane@example.com

User controls

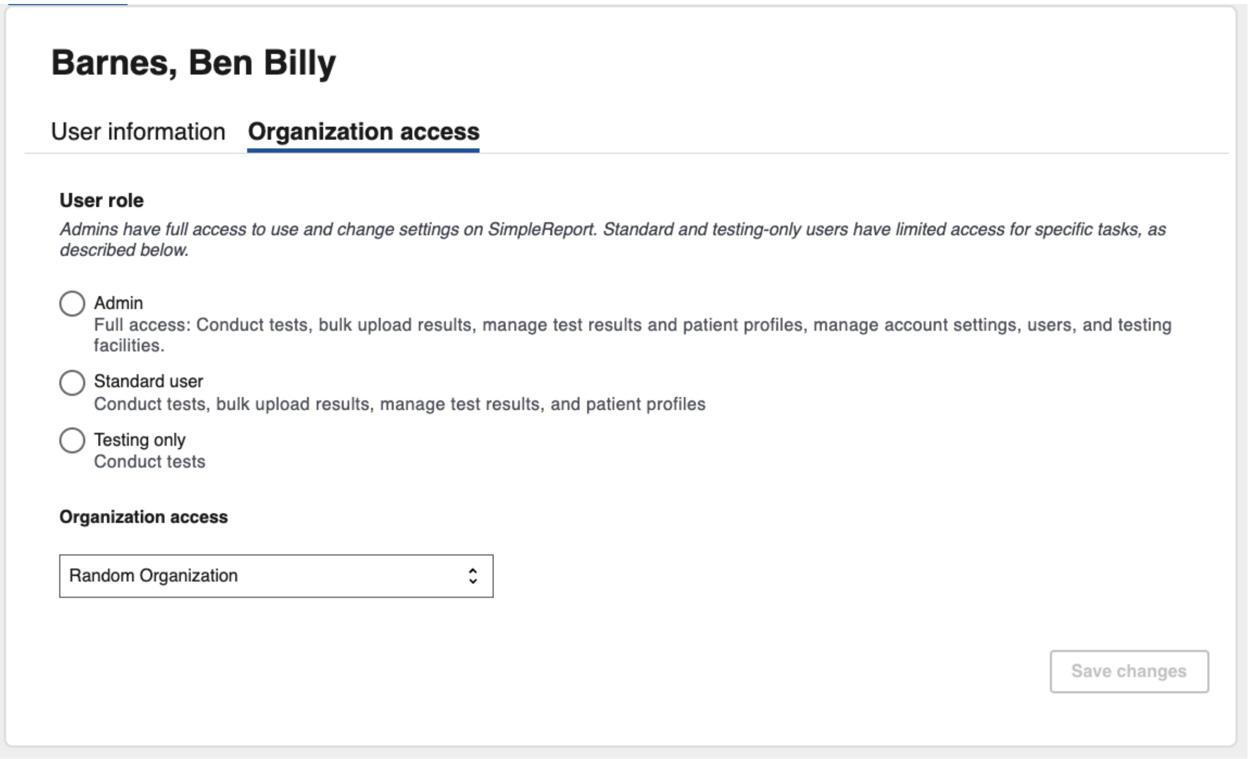
Password
Send a link to reset user password. Users must answer a password recovery question to access their account.

Reset multi-factor authentication (MFA)
Reset user MFA account access settings

Undelete user
Restore a deleted user account and data

Moving user to an organization

This feature will be available under the user view organization access section. The organization access will allow the support admin to update the role and organization of a user.



Barnes, Ben Billy

User information **Organization access**

User role

Admins have full access to use and change settings on SimpleReport. Standard and testing-only users have limited access for specific tasks, as described below.

○ Admin
Full access: Conduct tests, bulk upload results, manage test results and patient profiles, manage account settings, users, and testing facilities.

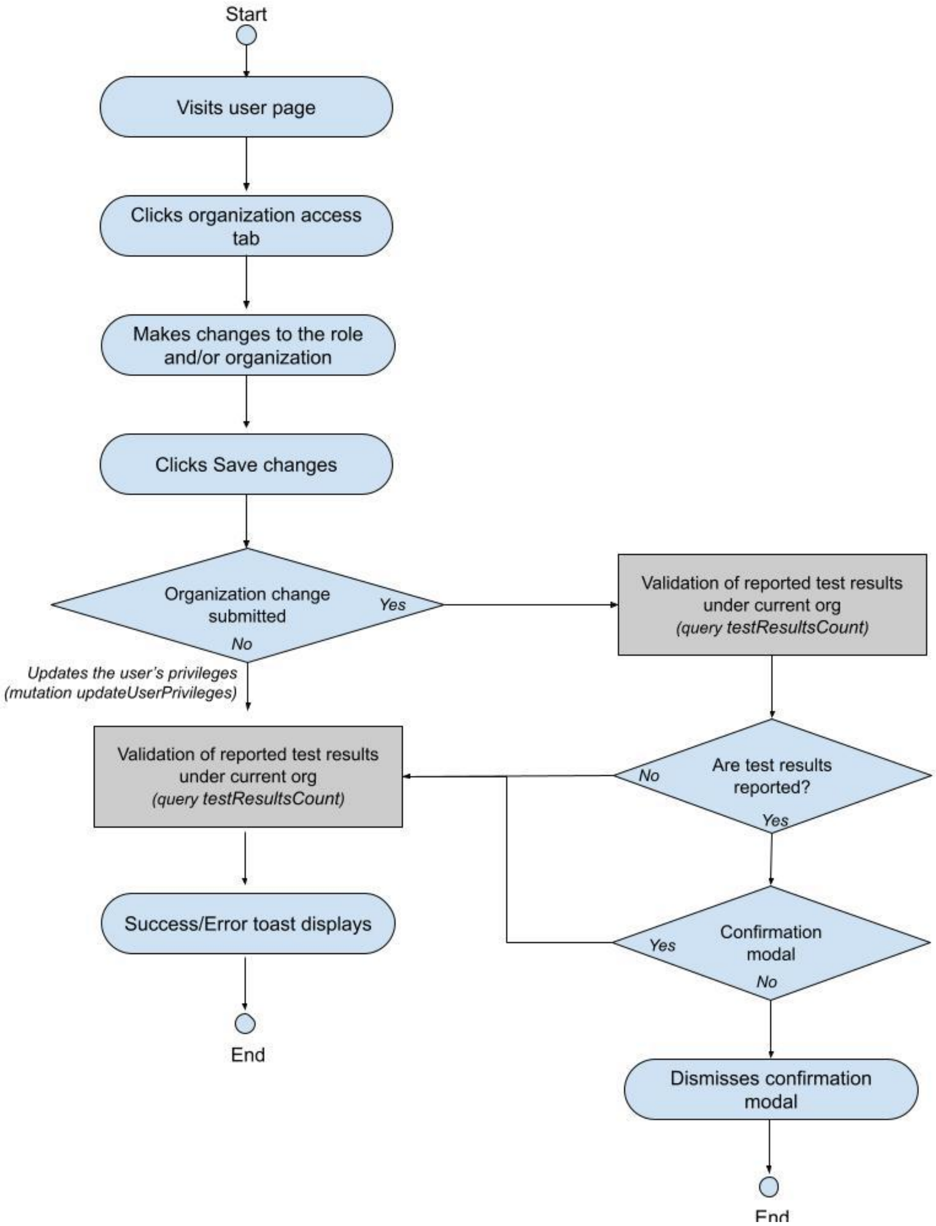
○ Standard user
Conduct tests, bulk upload results, manage test results, and patient profiles

○ Testing only
Conduct tests

Organization access

Random Organization

The suggestions in this section are based on the assumption that moving users to an organization flow will follow the steps shown in the below diagram. It also assumes that the action can be performed for non-deleted users only.


Start
Visits user page
Clicks organization access tab
Makes changes to the role and/or organization
Clicks Save changes
Organization change submitted
Yes
No
Validation of reported test results under current org (query testResultsCount)
Updates the user's privileges (mutation updateUserPrivileges)
Validation of reported test results under current org (query testResultsCount)
No
Are test results reported?
Yes
Success/Error toast displays
Yes
Confirmation modal
No
End
Dismisses confirmation modal
End

Organization access tab

The purpose of this tab is to allow support admins to update the roles and organization of an user. Before allowing an update of organization the application needs to check that there are no test results reported under the current organization where the user is currently assigned. The reason is that the moment the user moves to the new organization it will lose access to any data reported under the previous one. To perform this check the UI will need to call the query testResultCount upon loading. It is recommended that UI disables the organization dropdown to avoid changes until the testResult check is completed.

If it is determined that there are testResults reported then the UI should display a warning to the support admin calling out the loss of data and a suggestion to confirm with the user that the move to the new org is indeed the path forward. Another more restrictive solution is to block the change of organization completely but this could end up in support admins needing development assistance still.

Constraints

There is no operation that allows checking if there are test results reported under an organization. The operation testResultCount is currently used to check if there are test results reported under a facility or for a patient but does not support filtering by organization.

Suggestions

Update the query testResultCount to accept an organization Id in its parameters and return the number of results under that organization.

Show a modal with a warning after clicking on save changes only if the testResultCount endpoint detected results under the organization and the support admin made an org change in the dropdown.

Saving changes

After the support admin has updated the role and/or organization the save changes button will become enabled. Upon clicking of the button the mutation updateUsersPrivileges will be called to update the user’s access.

Constraints

The mutation updateUsersPrivileges, currently used in the Manage users page within settings, does not support updates of organizations. It only allows updates for roles and facilities so there is no operation available to move a user to an org.

The application also needs to make sure that before adding a user to the necessary okta groups for the new organization the user is removed from the previous groups.

Suggestions

We have two options:

1. Create a new endpoint to support the access changes. If we follow this path we would need to make sure that we can reuse the current logic to allow for role updates.
2. Reuse the updateUsersPrivileges endpoint to accept the updates on organizations. This option gives us the ability to update roles for free but will need us to add checks to make sure that Super Users are the only ones allowed to request for organization changes.

Alternate:

- Currently the createUserInCurrentOrg call does allow deleted users to be re added to the same organization. During this reprovision step we throw an error if the user's claims do not match the same organization. We could refactor this to allow this logic by simply removing the existing claims. *(This could be implemented for the organization admin -> undelete user flow so by d*efault putting back the user to their previous org as soon as they become undeleted but the organization admin cannot see deleted users so they won't be able to trigger the undelete action)

Open questions

How is this currently being done in the manage users page?
The user access is updated with the operation UpdateUserPrivileges. This operation only supports updating the role and/or facility access.

Does the current user update allow for organization update?
No. It only allows for facility access and role updates.
Is there an endpoint available to check if an organization has test results?
No. There is an operation testResultsCount that could be expanded to support search by organizationId.

If there are results under the current user's organization. Do we block the change organization action (by disabling the dropdown) or do we just show a warning stating that the user will lose access to the tests reported and to check with the user before performing the organization update?
TBD

After the user is moved to the new organization. Does it get assigned to all facilities by default? Or does it get assigned to no facilities and the org admin is expected to finish setting the user?
TBD

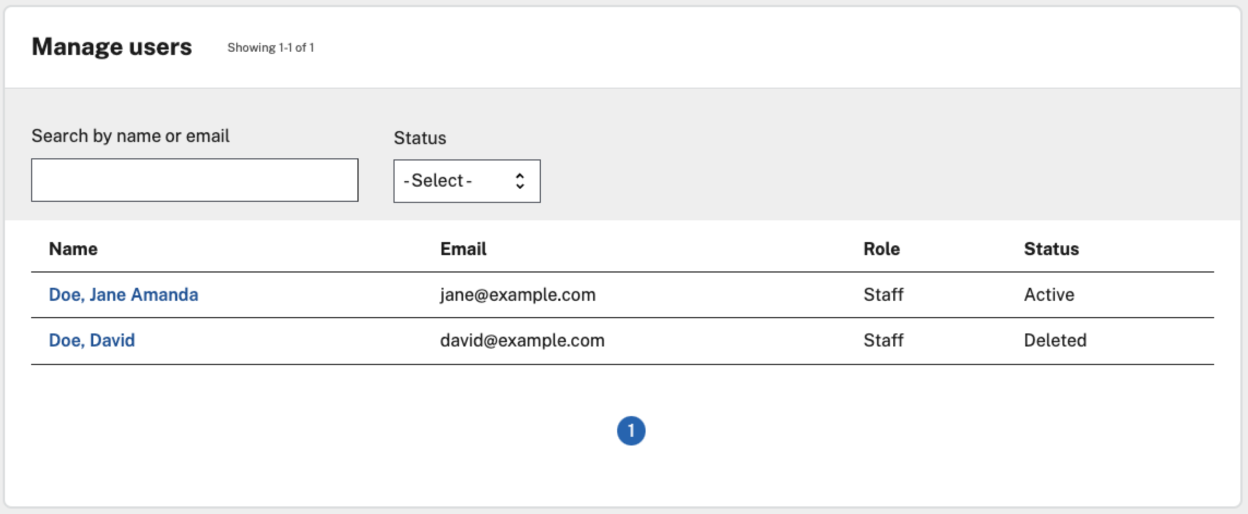
How does the add admin to organization flow add a new admin to an organization?
It uses the operation AddUser. But this operation creates an entirely new user in the application as part of its logic.

Open Questions
Can we include some breadcrumbs or any link that takes you back to the Support admin page.

Second Iteration on the manage users experience

Manage users

(This page will not be implemented in our first iteration of features for the support admin and will be revisited later as a concept)



Manage users Showing 1-1 of 1

Name	Email	Role	Status
Doe, Jane Amanda	jane@example.com	Staff	Active
Doe, David	david@example.com	Staff	Deleted

We need the ability to retrieve users across all organizations and to filter the retrieval based on their status, email and name. The operation also needs to support pagination.

The search triggers when the support admin clicks the search button (the search bar in this page will have the button displayed). Before triggering the search we should validate that the content in the field is still useful. For example it should have more than 3 characters. The search button will remain disabled if the field is not properly filled out.

Add organization as part of the column. (Name, Email, Organization, Role, Status).

The role maps to the user role (admin, standard user, testing only) mocks need to be fixed.

Another idea that came from the team review was to only allow search of users by email. With this path we don't need to support pagination and special filtering. Can we confirm that this would be the case?.

Constraints

- The current graphql operations that allow users retrieval (users, usersWithStatus) only gather users within an organization (logic based on the **getUsersAndStatusInCurrentOrg** method). They also do not support search filters, pagination, return of deleted users and they don't return the user's role.

- There are two account statuses that the application needs to start supporting: deleted (users soft deleted), recovery (users in the middle of resetting their password/reactivating their account)

Suggestions

To retrieve the necessary information to populate the table we have two options:

1. Create a query operation specifically for the support admin role within admin.graphql.
2. Modify the current operation usersWithStatus to return users across all organizations if the role of the user requesting the data is a super user (support admin). This could probably be done by checking the ApiUserContextHolder object in the service method that resolves the request.

Both options will still need effort to allow for:

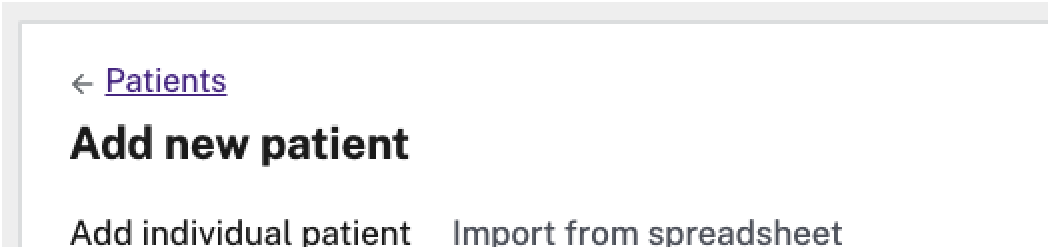
- Pagination
- search filtering
- inclusion of deleted users in the results
- Return the role, status and isDeleted information per user

Probably we should include a modal that opens an informative table with all the different account statuses explained.

At the end the team decided that a manager users page was not necessary for a first round of implementation. The support admin will have a Manage user page instead that will display the user view and a search by email field at the top.

Notes

- If this page gets implemented the user view page will need a link at the top that takes it back to the managed users page “ Go back to Manage users”. Example in the add individual patient page.



← Patients

Add new patient

Add individual patient   Import from spreadsheet

Answered questions

Does *UsersAndStatus* endpoint allow pagination?

No

Does *UsersAndStatus* allow filtering of results?
No

Does *UsersAndStatus* return deleted users?
No. Currently the query operations that return a list of users, either users or usersWithStatus, do not allow retrieval of deleted users.

What does *UsersAndStatus* return?
Users object with limited fields that do include the okta status. However, the query does not allow retrieving deleted users and even if it did it is not possible to detect a deleted user with the information returned.

Do we need to include a DOB in the *Manage users* page?
We don't as the email information is used as a unique identifier with okta.

Do all types of account status (*deleted, undeleted, active, suspended, stage, provisioned*) are returned as part of the user object?
Currently the operation usersWithStatus does return the okta status in the status field. The query users do not return the okta status of the account.

The query user returns the okta status information in the status field. However, the application does not support the retrieval of deleted *api users.* The deleted api user state will need to be represented with a combination of a status = suspended and some sort of new field to track if they have been soft deleted in our database. This is due to the fact that the okta state of suspended is currently being used to **represent deactivated and deleted users** in the application.

Because the UI is not currently handling any operations with deleted users this has not been an issue but with the new features coming for the support admin a differentiation needs to be made so the UI can present the proper actions for deleted api users.

Okta	**Simple Report UI**	**Notes**
Suspended	**Deleted (Not used)**	This status is not properly represented in the data due to sharing the okta state with deactivated status
Suspended	Deactivated	
Provisioned	Pending	

Deactivated	Not used	This status removes the passwords in the okta account.
Activated	Active	
Recovery	**Active** *Used but should be displayed as Recovery*	When someone is in the process of password reset. Surfacing this to the org admin could be taken later but it will be shown to the support admin.
Staged	Not used	This state is pre-provisioned and used to happen before the sign up flow was implemented.

Should we surface any information related to the okta groups that exist in the table or does the role suffice? Example State:Org_id_ ALL_ACCESS
I think the role information is enough for a support admin to set the correct access for an api user. The application should keep handling the okta group setup to abstract its complexity to the support admin.