

View permissions (Super admin)	As a super admin, I want to be able to view the roles and permissions on the verve portal so that I can be well informed about the roles and permissions on the verve portal
---------------------------------------	--

Audit Log	As a Super Admin/Corporate Admin/Verve admin, I want to be able to access and review the audit trail to track and monitor user activities, system changes, and important events on the platform for security, compliance, and accountability purposes.
------------------	--

Deactivate Verve Admin - Super Admin	As a Super admin, I want to be able to deactivate a Verve admin, so that he /she does not have access to the Verve portal.
---	--

1. Super admin must be able to see a roles and permission feature from the menu list.

2. Super admin must be able to see a dropdown list of available roles.

3. Once a role is selected, super admin must be able to see the permissions tied to the role selected.

1. The "Audit Trail" section should display a comprehensive log of all significant activities, events, and changes that occur on the platform, such as user logins, user logouts, role changes, and system settings modifications.
2. Each entry in the audit trail should include details such as the timestamp of the activity, the user or admin responsible for the action, the type of activity performed, and any relevant metadata.
3. The audit trail should capture logs for both Verve admin and other users
4. The audit trail should be easily searchable and filterable based on various criteria, such as date range, specific users, or types of activities, to facilitate quick and efficient data retrieval.
5. An admin should be able to export the audit trail data in various formats (e.g., CSV, Excel, PDF) for further analysis and reporting.
6. The audit trail should record not only successful actions but also any failed attempts or errors for comprehensive tracking of user activities.

"

option. (This exists).

Super admin must be able to see and select 'My Users' tab to see the list of Verve admins created. (this exists)

Super admin must be able to see and hover on the menu icon on the 'Action' header (this exists).

If the Verve admin's status is active, Upon selecting the option above, Super admin must be able to see the following options:

Edit- to edit user's information. (this exists)

Manage application- to manage user's application. (this exists).

Deactivate user- to deactivate the user's profile on Verve portal.

Upon selecting the 'deactivate option', super admin must be able to see a comment box to input reasons for deactivation and submit.

Super admin must be able to confirm their submission.

Super admin must see a successful pop-up notification.

The Verve admin deactivated must not be able to login to the Verve portal.

The Verve admin's status must be displayed as inactive on the table.

Super admin must be able to filter by status - Active/Inactive

Verve admin must receive an email notification when deactivated.

FE:

- Grant super admin access to view roles management

BE:

- Refactor the existing service to retrieve roles from the database to accomodate super admin
- Handle edge cases where no roles exist or the user does not have the required permissions
- Refactor the existing service to retrieve the permissions associated with a specific role from the RolePermissionMap table
- Ensure the API checks that the super admin is authorized to view the roles and it's permissions

FE:

- Create an Audit Trail Table
- Add a search input field to search by app name
- Add filters for actions, status, and date
- Implement download/export options (CSV, PDF, Excel)

BE:

- Creation of an endpoint to fetch the audit trails and adding the search and filter feature.
- Creation of an endpoint to download the report of the audit trail in the pdf/csv/excel format.
- Integrate the audit service to the existing endpoints.
- Ensure the audit service processes successful and failed actions.

BE

- Update the users table in the DB to accomodate new columns
ActiavtedOn(When the user was activated or deactivated)
ActiavtedBy(Who activated or deactivated the user)
Comment (for storing activation/deactivation reasons)
- Implement API functionality to handle user activation/deactivation, including changing their status to Actice/Inactive
- Update the API endpoint to accept a "reason for activation/deactivation" input and log it in the database.
- Add a check in the authentication logic to block login attempts by users with "Inactive" status.
- Update the logic to ensure that activated users appear as "Active" and deactivated users appear as "Inactive" in relevant API responses.
- Integrate an email service to notify Users when their status changes to "Inactive," including the reason if applicable.
-

FE

1. Add Deactivate User Option to the list of Super Admin dropdown
2. Create a modal to allow super admin input deactivation reasons
3. Create a confirmation modal upon super admin submission of deactivation reason
4. Display a message to the super admin upon successful or unsuccessful deactivation