# QA-Shared cabinets test cases

Types of users- collaborators, invitees, banned users, owner

Module- invite-members

1. If owner-controlled only owner can invite. Collaborator is NOT allowed to invite.

a. Only users with syncweave permissions can be invited.

b. Test that all other users (w/o syncweave permissions) fail.

c. More than one user invitation at a time should work fine.

d. No invitee entered while sharing should work fine.

e. Owner/collaborator inviting himself.

f. Inviting banned users should NOT be allowed.

g. Sending more than one invitation to the same invitee. User should not appear more than once in the invitee list. User shouldn’t receive multiple emails.

h. Invite a user who has been kicked out.

i. When the invited user accepts the invitation

a. fail and dismiss request if cabinet has been unshared.

b. succeed if owner has changed.

c. fail and dismiss if original inviter has been banned.

2. If not owner-controlled above tests (a-i)should be run against collaborators too.

Module- Request\_ownership

1. Collaborators should be allowed to request ownership.

2. Non-related users, banned users, owner and invitees should NOT be able to see this option.

3. More than one 1 collaborator requests ownership

A is owner. B requests ownership so does C. A approves B. Can C approve C’s request? No

The request will continue to show up on A’s request page if A presses “accept” they should see an error mssg that the request is no longer valid.

Module- Offer-ownership

1. A is owner B requests ownership. A approves & offers ownership to B. Make sure of the following

a. A should now have collaborator rights i.e

- He should be able to invite other users (is user-controlled is false)

- He can request to be an owner again.

- Permission to “request\_member\_invite” (if user-controlled true)

- leave,meta-data,mount,umount,list\_shares – all should be allowed

b. A no longer has “owner” perms

-no longer has the rights to approve owner requests

- can no longer kick-out users

-can no longer ban users

- can no longer unshared

2. A is the owner. B requests ownership. Before A accepts ownership

A gets deleted. No problem.

B gets deleted. When A trasfers ownership should see an error message “user no longer exists”

Edge cases / negative test cases

1. Owners/Collaborators/Invitees getting deleted by admin when they are involved in an active transaction.

2. Invitees not having enough quota space after accepting invitation (handled in detail later)

3. Invitees accepting invites after the owner has changed/banned/kicked-out (old invitations will stand unless the old inviter is banned)

4. Duplicate file names. Invitee accepting invitation to shared folder “x” where he already has folder “x” in his cabinet.

Module- kick-out users

1. Make sure kicked-out users no longer see the shared folder in their cabinet and under the shared folders list.

2. kicked out users should be allowed to become collaborators again.

3. Owner cannot kick out himself.

4. Collaborator gets deleted after he sends request to invite. Does it get approved If owner controlled- YES If not owner-controlled- YES

Module- ban\_users

1. Owner cannot ban himself.

2. Owners should be allowed to ban invitees those who haven’t accepted the invitation yet

3. When such users (2 above) accept the invitation they should see an error message.

4. banned users should no longer see the shared folder in their cabinet and under the shared folders list.

5. If B was previously a collaborator and requested ownership before he got banned. What shouls happen if the owner by mistake transfers ownership? Error message & proposal is retained afterwards.

6. B was a collaborator before getting banned. He sent out invites before getting banned. Should those invites still work after he gets banned. No, they will fail and then be dismissed.

Module- unsharing

1. Collaborators, invitees, banned and other non-related users do not get to see the “unshared” option.

2. Unsharing should make the folder disappear from the collaborator’s and owner’s cabinet& share list.

3. What happens if the owner unshares while a collaborator is uploading something to DB? The uploaded files should not be lost. All other shared folder contents should disappear .

4. What happens to all unaccepted/unapproved invitations/owner requests?

a. When invitees accept the invitation.

b. When owners approve ownership requests

Error message followed by proposal dismissed.

Module- Leave shared folder

1. Owners should not see the “leave” option before they unshared/transfer ownership.

2. Collaborator B invites C who still hasn’t accepted the invitation. B leaves. What happens when C accepts? Will work as long as the original inviter has not been banned.

3. Only collaborators should see the “leave” option.

4. After “leaving” the folder should disappear from their cabinet and sharing page. Owner should not see this user in the cabinet members list. This user can be banned/re-invited to the shared folder.

Testing valid and invalid Mount locations

1. /foo/car/trunk

/boo/ zoo

User decides to mount at /foo/\_ where “boo” is already a shared folder. Invalid.

2. User decides to mount at /foo/boo/\_ where “boo” is already a shared folder. Valid.

3. User decides to mount at /\_ where “foo” is already a shared folder. Valid.

4. Converting “foo” to a shared folder where “boo” is already shared. Invalid.

5. Converting “zoo” to a shared folder where “car” is already shared. Invalid.

6. Test that moving an existing shared folder underneath an existing shared folder is invalid.

Module- metadata

1. Owner should be able to list metadata about share folder.

2. Collaborator should be able to list metadata about shared folder.

3. Non-related users, kicked out and banned users should not have this option.

4. Invitees should see basic info about shared folders.

\*\*Make sure that in all test cases above the shared folder details displayed are correct (name, no.of files/folders etc.)

Module- mount

1. Owner should not see the mount option.

2. All collaborators should be able to see the mount option on the shared folder page. (If the folder is not already mounted. )

3. If the folder has not been mounted, it should not show any current mount location.

4. If the folder has been mounted it should show the current mount location & an “unmounts” option only.

5. Make sure that really long mount locations are displayed w/o any problems.

6. Make sure the collaborator can mount the folder in the following locations.

a) root level

b) 2 level deep

7. To make sure that mounting does not break other file cabinet functionality:

a. \*\*test all test cases from cabinet general test cases doc.

b. Shared folders do not have the “delete” option. They can only be unmounted, moved, make local copy & renamed.

c. Copy the shared folder. – Create local copy should be allowed. If the same name used for the copy error message should show.

d. Renaming the shared folder- Changes should be visible only to the action initiator in his cabinet only.

e. email-files under shared folder- should work the usual way.

8. Mounting a shared folder at a location that already has a folder with the same name as that of the shared folder.

Make sure a (1) is appended to the shared folder name.

Make sure that this change in the shared folder name is only visible to the action initiator & not to other collaborators & owner.

Module- unmount

1. Make sure that unmounting gets rid of the shared folder from under the file sync page. This should not affect other collaborator’s and owners mount location (if any)

2. Shared folder list page should now show a “mount” option.

3. Mount location should be blank.

4. Check file sync page actions to make sure that “unmounting” didn’t break anything.

\*\*Test general cabinet file and folder actions from general cab actions doc.

Module- create\_share

1. Make sure that “creat-share” is successful only if the user has syncweave permission.

2. Create-share from nothing- User should be given the option to create a new shared folder.

This new empty folder should appear on the shared folder list view & the owner should see a “mount” option.

The mount location should be blank.

3. Creating a shared folder by converting an existing folder-

-User with syncweave permission should see the “convert to shared folder” option when a single folder is selected on file sync page.

-After converting the regular folder to a shared folder this folder should now appear on the “shared list” page with the “unmounts” option & the current mount location displayed.

Module- List-shared

1. Make sure that the following users can see/use the list\_shares option.

-users w/ syncweave permission & have no shared folders This page should be blank.

-users w/ syncweave permission, is owner to one folder and collaborator for another.

- users w/ syncweave permission & is only an owner

- users w/ sync permission & is only a collaborator