TRYCLOUD

# Selenium-TestNG framework assignment

Company website : https://trycloud.net/

Cybertek

## Description

In this assignment each group will create an automated testing framework for testing TryCloud company’s application.

The framework will use : Java
Maven
Selenium WebDriver
TestNG
Git/Github

Discuss the “**git branching strategy**” first, then implement it during the automation.

Conduct “Code Review” whenever is needed to explain any re-usable method to the team, or go over **PR.**

Create one class for each user story.

**Application information:**

Credentials link: [click here]

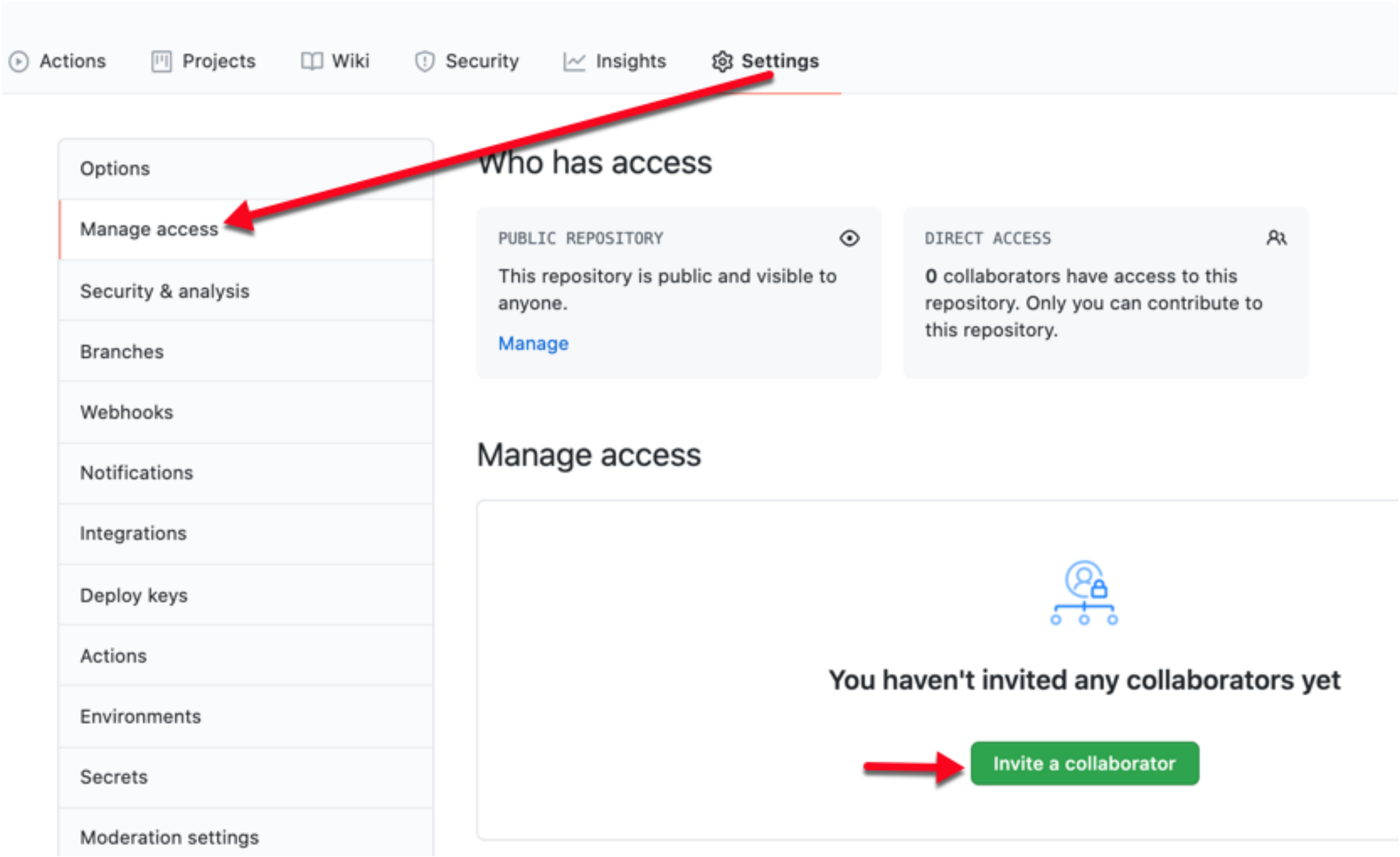
## Things to do:

One person from the team :

1. **Set up the project** in his/her environment while the other team members are watching (Follow the next page)

2. Create local repo

3. **Create a remote repository** & **add** all the **other** team members **as collaborators**



Who has access

PUBLIC REPOSITORY
This repository is public and visible to anyone.
Manage

DIRECT ACCESS
0 collaborators have access to this repository. Only you can contribute to this repository.

Manage access

You haven't invited any collaborators yet

The other team members :

1. Accept the GitHub invitation
2. Clone the repository

# Project set up

1. Create a maven project.

   Group-id: **com.trycloud**

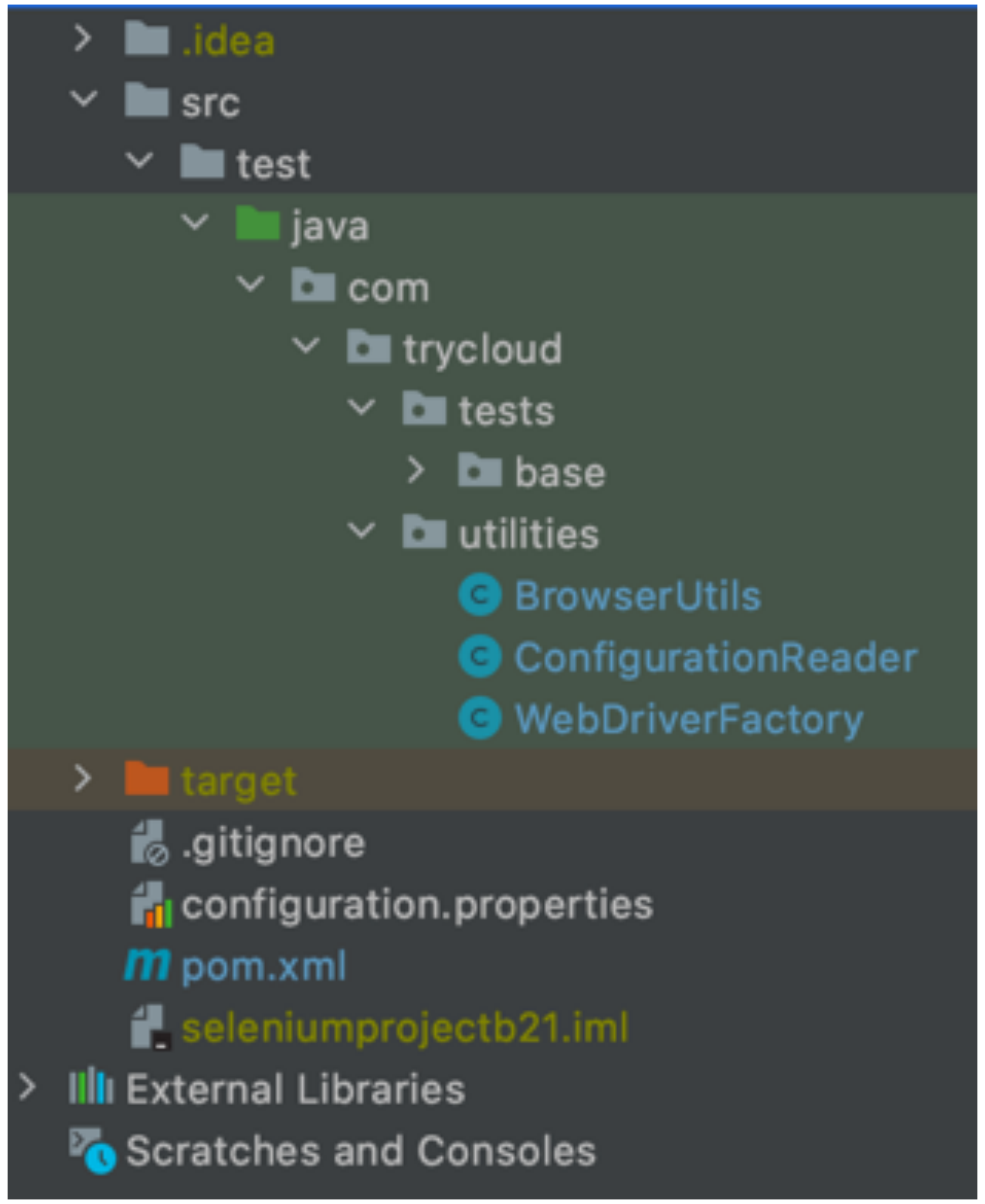
   Artifact-id:  **trycloudautomation**

2. **Add** all the required **dependencies** for automation to the **pom file**.

3. Under src/test/java create following packages:
   a. com.trycloud.tests
   b. com.trycloud.tests.base
   c. com.trycloud.utilities

4. Under the project folder add file configuration.properties
   add file .gitignore

5. Under utilities package create:
   a. ConfigurationReader
   b. BrowserUtils
   C. WebDriverFactory



```
> .idea
v src
  v test
    v java
      v com
        v trycloud
          v tests
            > base
          v utilities
              BrowserUtils
              ConfigurationReader
              WebDriverFactory
> target
  .gitignore
  configuration.properties
  pom.xml
  seleniumprojectb21.iml
> External Libraries
  Scratches and Consoles
```

NOTE: This picture shows what packages you should have up completion of steps 1 to 5.

1. **Story**: As a user, I should be able to login.

### Test case #1 - verify user login successfully

1. Navigate to login page http://qa.trycloud.net/index.php/login?clear=1
2. Enter valid username
3. Enter valid password
4. Click login button
5. Verify the URL is chawed to homepage’s url

(Put this test case in testBase class with an annotation so it runs for all the test case)

### Test case #2 - verify user cannot login with invalid credentials

1. Navigate to login page http://qa.trycloud.net/index.php/login?clear=1
2. Enter valid username  invalid password
3. Click login button
4. Message “Wrong username or password.” should be displayed
5. Page title and url should be same

2. **Story**: As a user, I should be access all the main modules of the app.

## Test case #1 - verify users access to the main modules

1. Login as a user
2. Verify the user see the following modules:

   **Files**

   **Galleries**

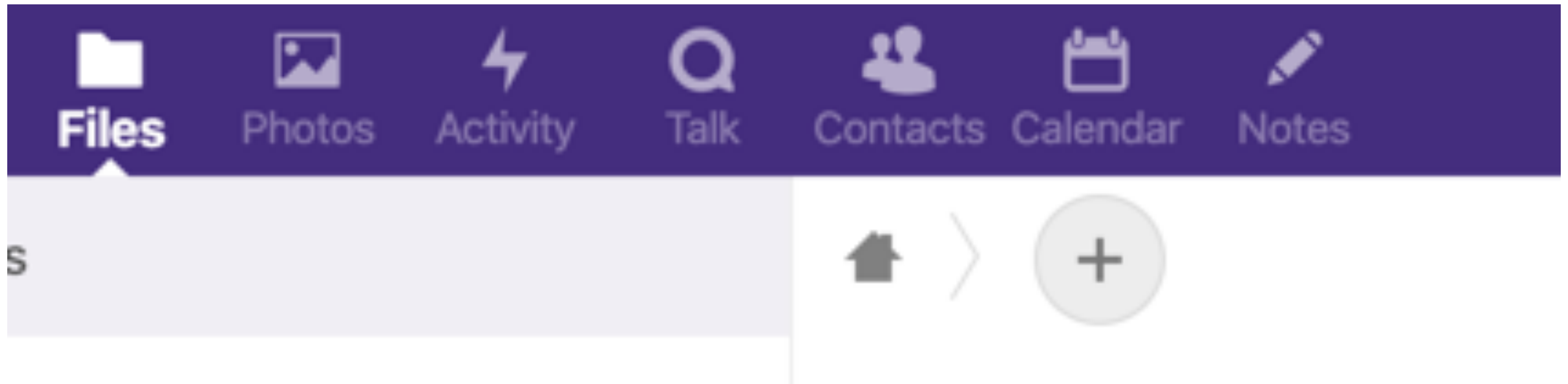
   **Activity**

   **Talk**

   **Contacts**

   **Calendar**

   **Note**



( Expected module name: Galleries
Actual module name : Photos )

3. **Story**: As a user, I should be able to access to **Files** module.

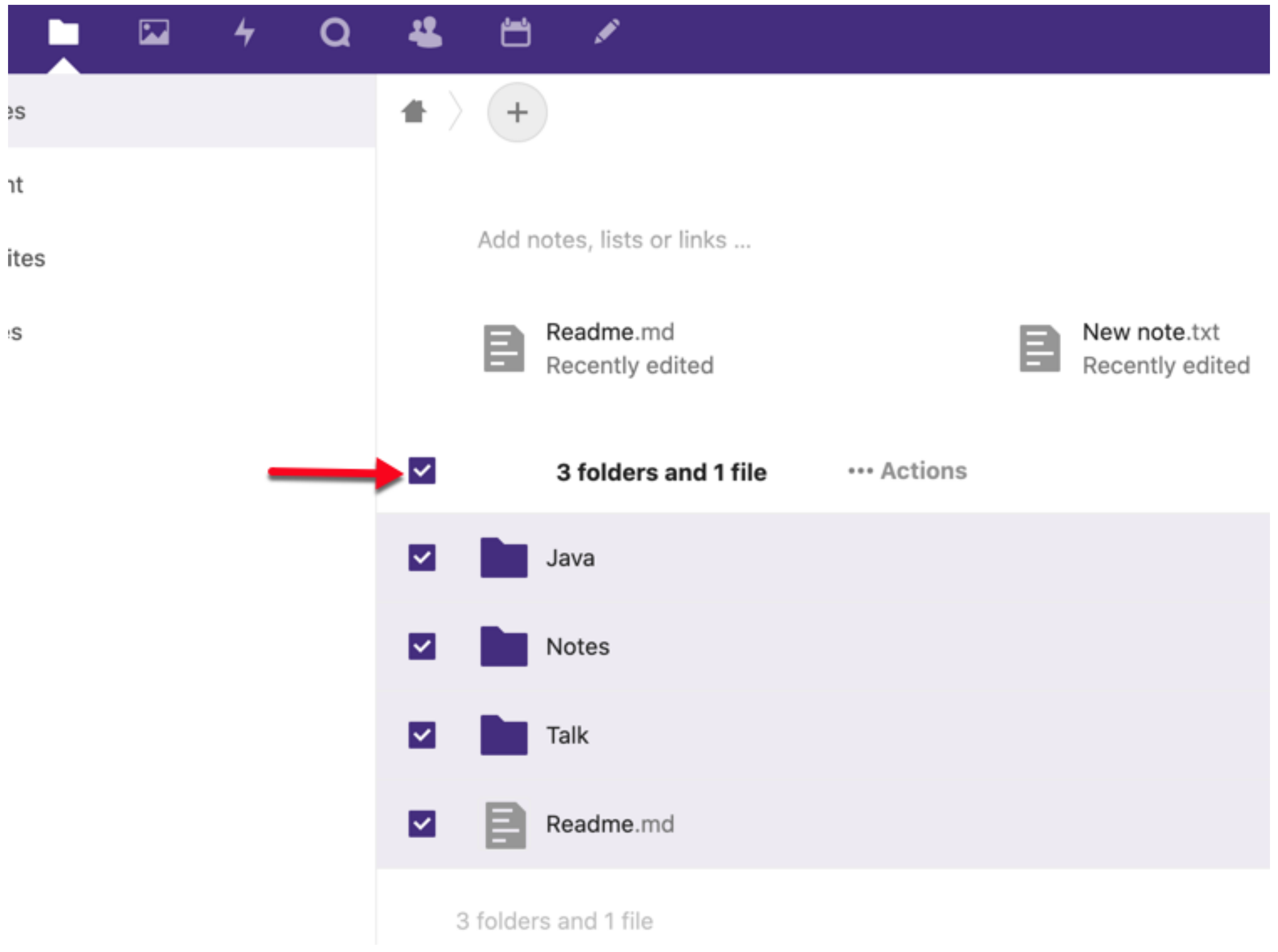
## Test case #1 - verify users can access to Files module

1. Login as a user
2. Verify the page tile is Files module’s tile

## Test case #2 - verify users can select all the uploaded files from the page

1. Login as a user
2. Click the top left checkbox of the table
3. Assert all the files are selected

(Pre-condition: there should be at least 2 files are uploaded the page)

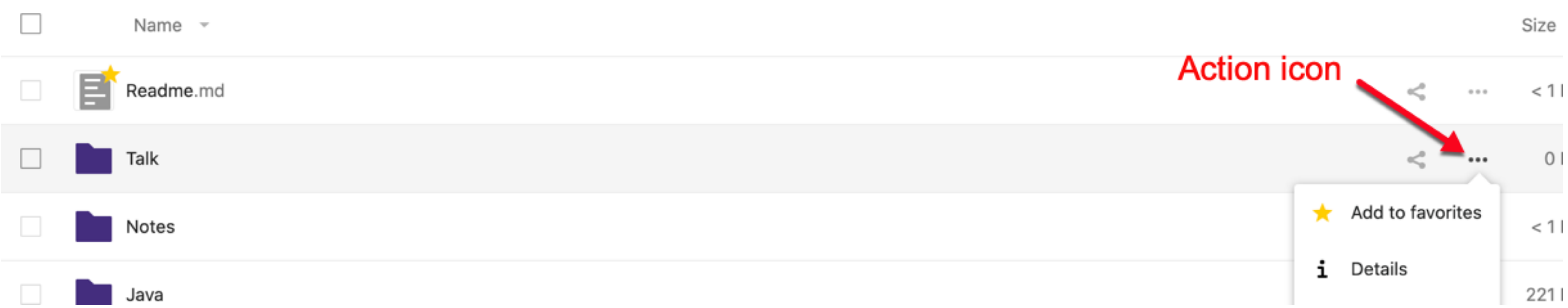


3. **Story**: As a user, I should be able to access to **Files** module.

## Test case #3 - verify users can add a file to favorites.

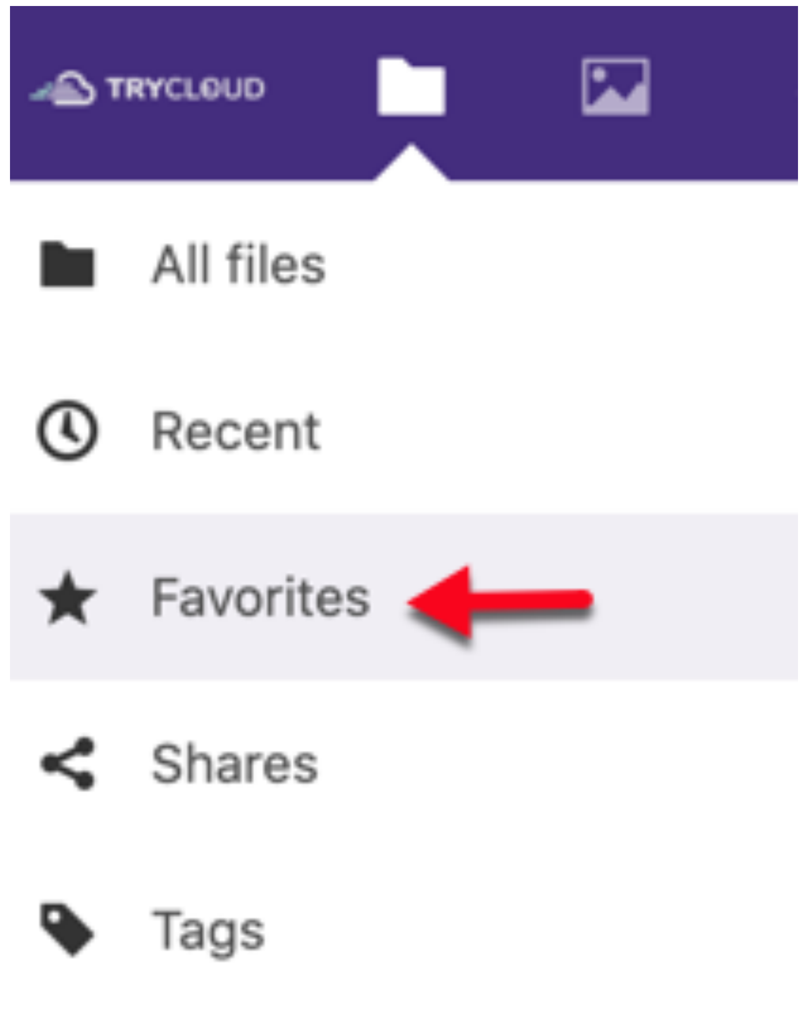
1. Login as a user
2. Click action-icon from any file on the page
3. Choose “Add to Favorites” option
4. Click “Favorites” sub-module on the lest side
5. Verify the chosen file is listed on the table

(Pre-condition: there should be at least 1 file is uploaded)



| ☐ | Name | Size |
|---|---|---|
| ☐ | Readme.md | < 1 |
| ☐ | Talk | 0 |
| ☐ | Notes | < 1 |
| ☐ | Java | 221 |

NOTE:this is for step 2



NOTE:this is for step 4

3. **Story**: As a user, I should be able to able to access to **Files** module.

### Test case #4 - verify users can remove a file from favorites.

1. Login as a user
2. Click action-icon from any file on the page
3. Click “Remove from Favorites” option
4. Verify that the file is removed from Favorites sub-module’s table.

(Pre-condition: there should be at least 1 file is added to favorites table)

### Test case #5 - verify users can upload a file directly to the homepage (File-Uploading)

1. Login as a user
2. Click the “+” icon on top
3. Click “upload file”
4. Upload a file
5. Verify the file is displayed on the page

3. **Story**: As a user, I should be able to access to **Files** module.

### Test case #6 - verify users can create a folder

1. Login as a user
2. Click the "+" icon on top
3. Click "New Folder"
4. Write a folder name
5. Click submit icon
6. Verify the folder is displayed on the page

### Test case #7 - verify users can upload a file inside a folder

1. Login as a user
2. Choose a folder from the page
3. Click the "+" icon on top
4. Click "upload file"
5. Upload a file
6. Verify the file is displayed on the page

(Pre-condition: there should be at least 1 folder is created on the filers page)

3. **Story**: As a user, I should be able to access to **Files** module.

## Test case #8 - verify users can delete a file/folder.

1. Login as a user
2. Click action-icon from any file on the page
3. Choose “delete files” option
4. Click deleted files on the left bottom corner
5. Verify the deleted file is displayed no the page.

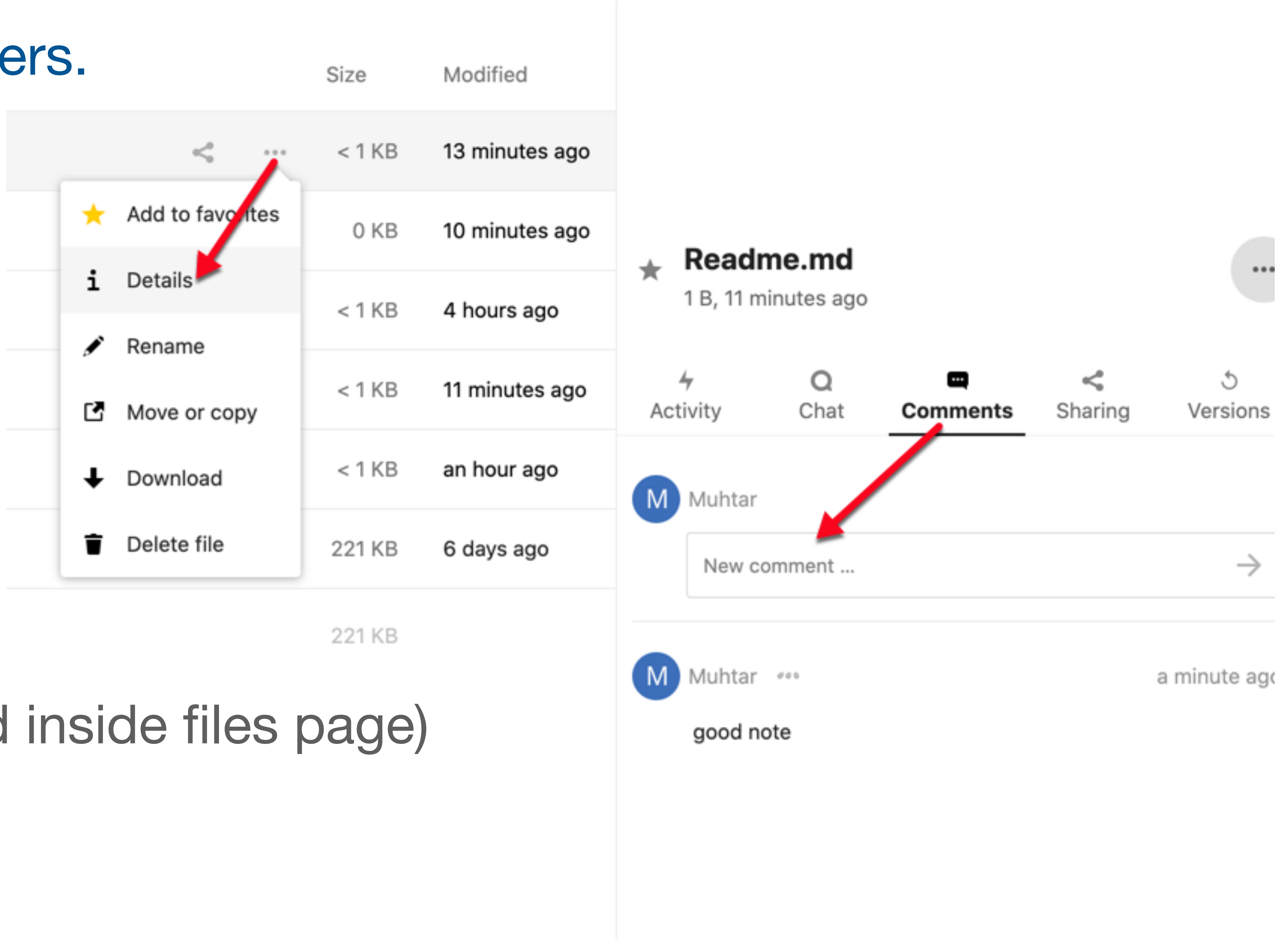
(Pre-condition: there should be at least 1 file is uploaded inside files page)

3. **Story**: As a user, I should be able to access to **Files** module.

## Test case #8 - verify users can write comments to  files/folders.

1. Login as a user
2. Click action-icon from any file on the page
3. Click details
4. Write a comment inside to the input box
5. Click the submit button to post it
6. Verify the comment is displayed in the comment section.

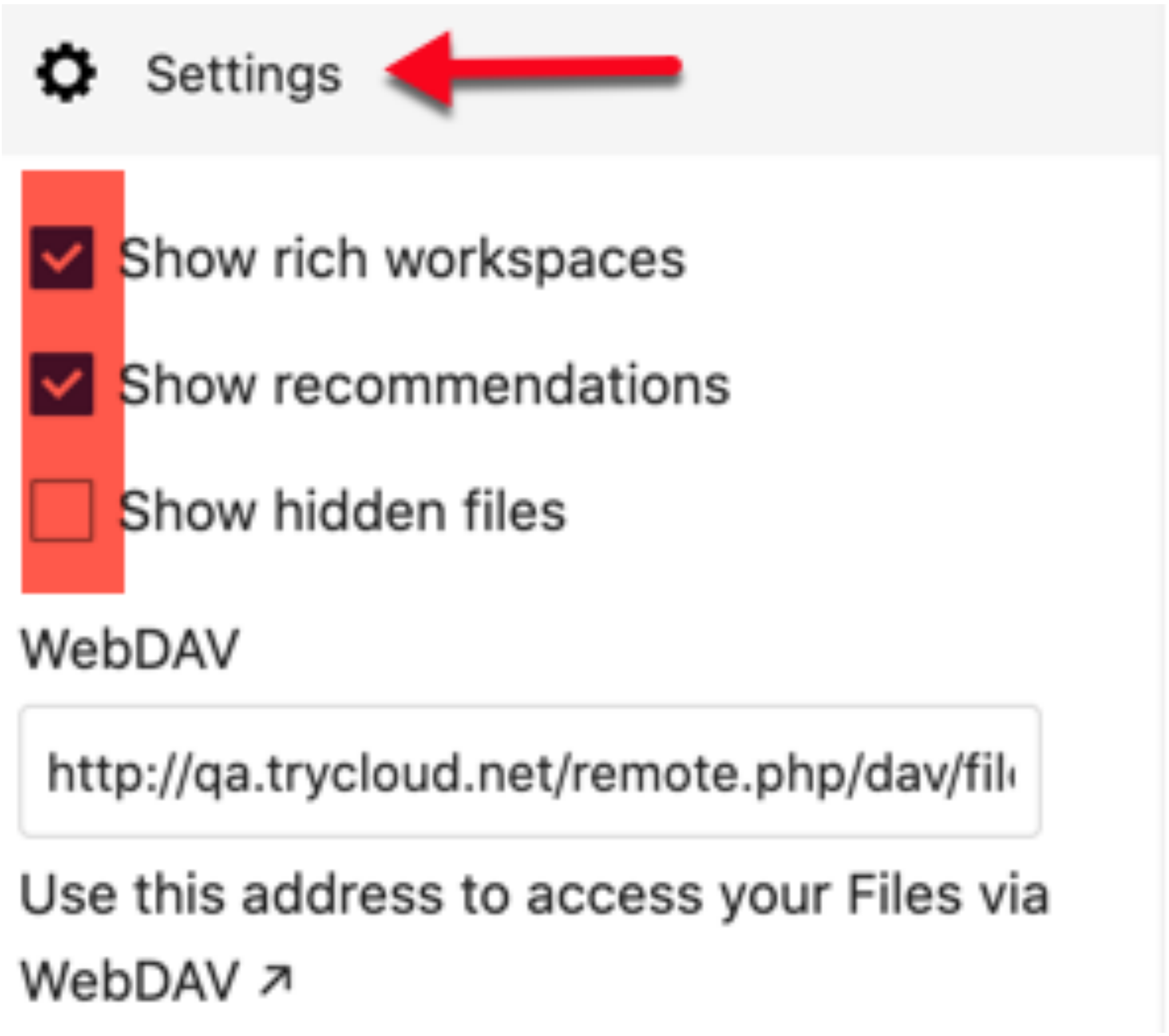
(Pre-condition: there should be at least 1 file/foler is uploaded inside files page)



3. **Story**: As a user, I should be able to access to **Files** module.

## Test case #9 - verify users can change the app Settings.

1. Login as a user
2. Click Settings on the left bottom corner
3. Verify user can click any buttons.



Settings

☑ Show rich workspaces

☑ Show recommendations

☐ Show hidden files

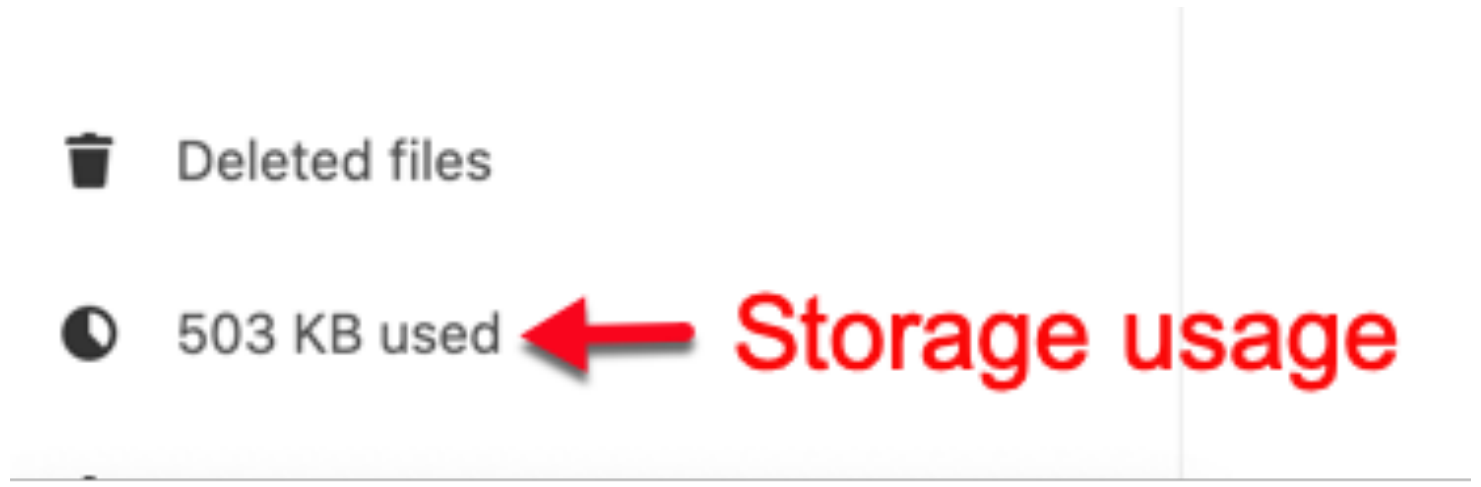
WebDAV

http://qa.trycloud.net/remote.php/dav/fil

Use this address to access your Files via WebDAV ↗

## Test case #10 - verify users see the app storage usage

1. Login as a user
2. Check the current storage usage
3. Upload a file
4. Refresh the page
5. Verify the storage usage is increased



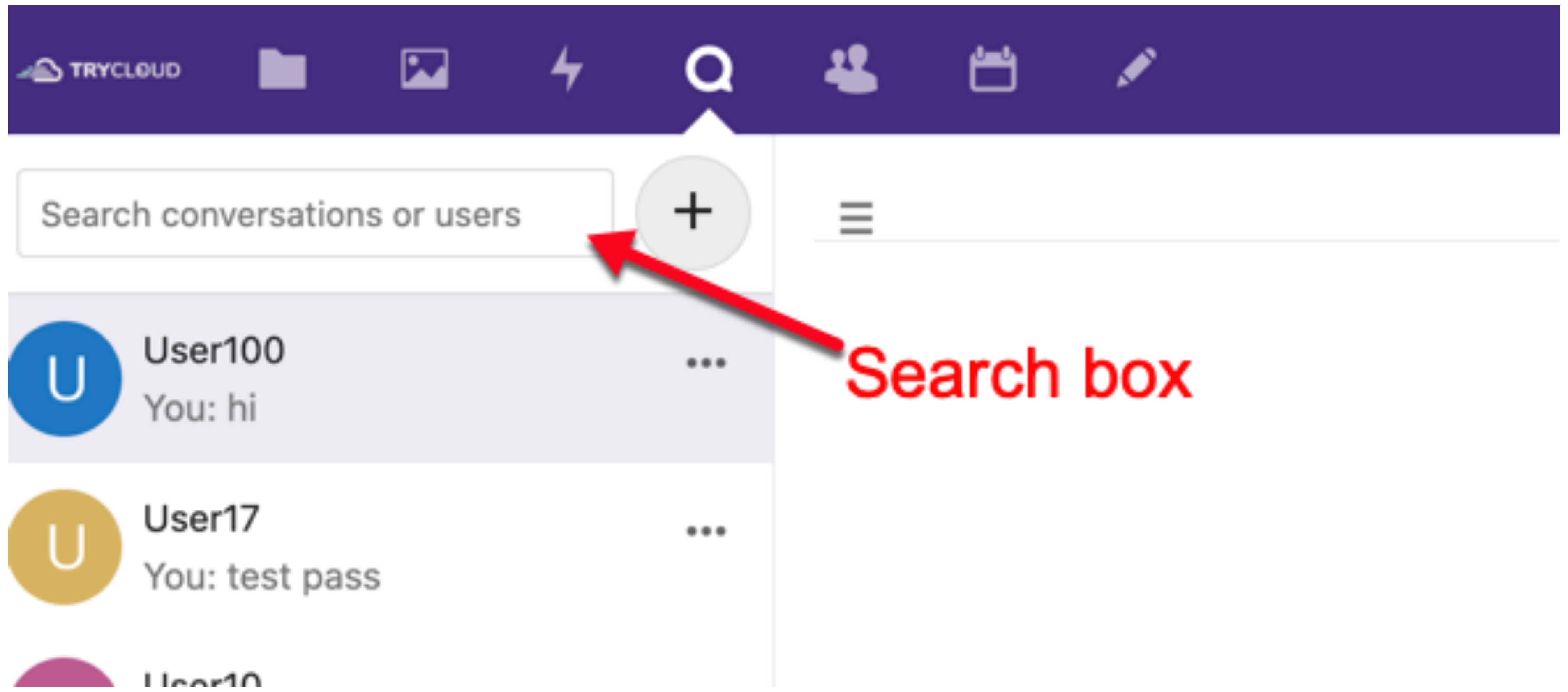
4. **Story**: As a user, I should be able to access to **Talks** module.

## Test case #1 - verify users can access to Talks module

1. Login as a user
2. Click Talks module
3. Verify the page tile is Talks module's tile

## Test case #2 - verify users can send message

1. Login as a user
2. Click Talks module
3. Search a user from search box on the left
4. Write a message
5. Click submit button
6. Verify the message is displayed on the conversation log



Search box

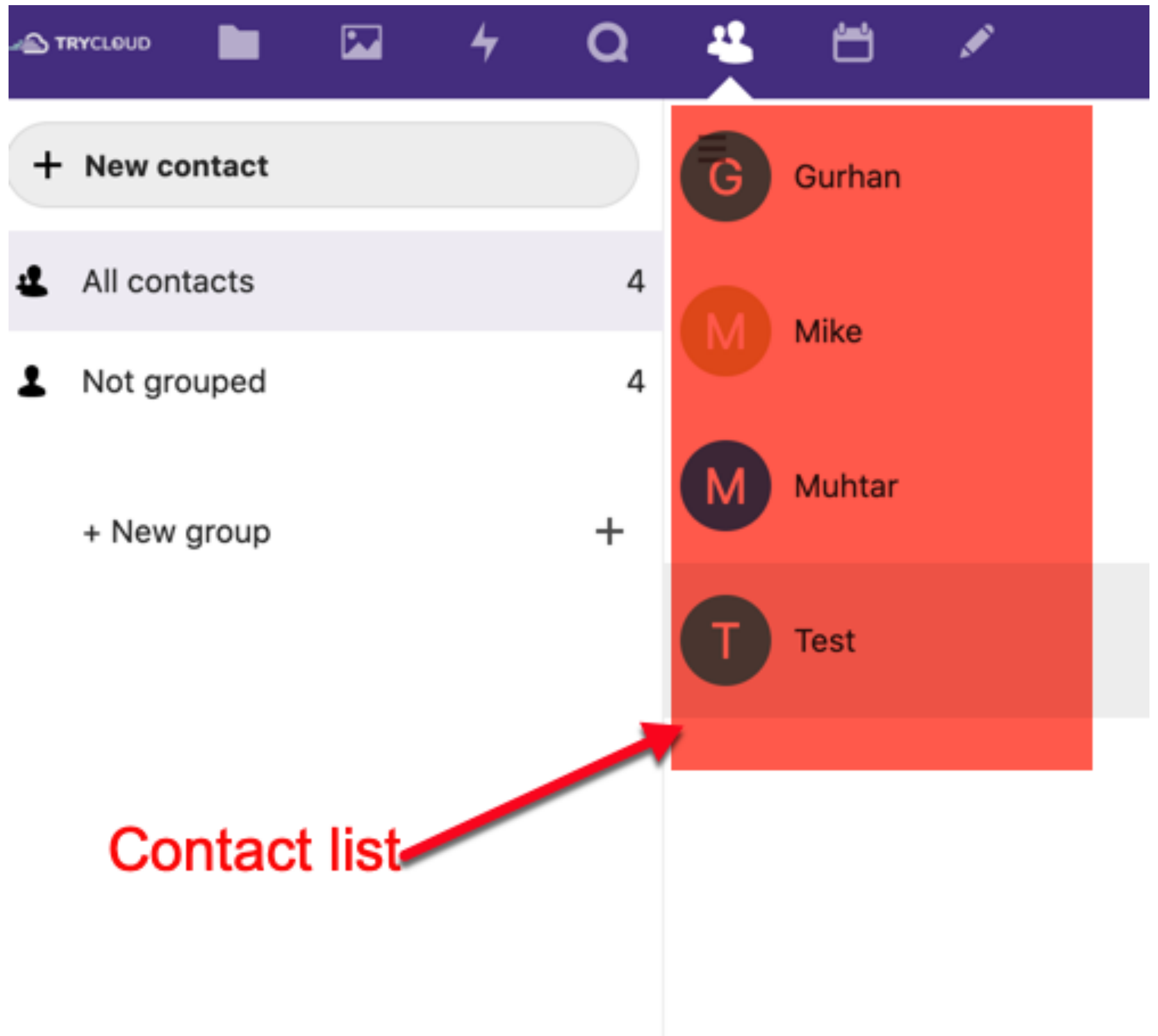
5. **Story**: As a user, I should be able to access to **Contacts** module.

## Test case #1 - verify users can access to Talks module

1. Login as a user
2. Click “Contacts” module
3. Verify the page tile is Contents module’s tile

## Test case #2 - verify users can add contacts

1. Login as a user
2. Click contacts module
3. Click “New Contact” button
4. Fill out the contact info like : Title, Phone, email, address , etc
5. Verify the contact name is added to the contact list

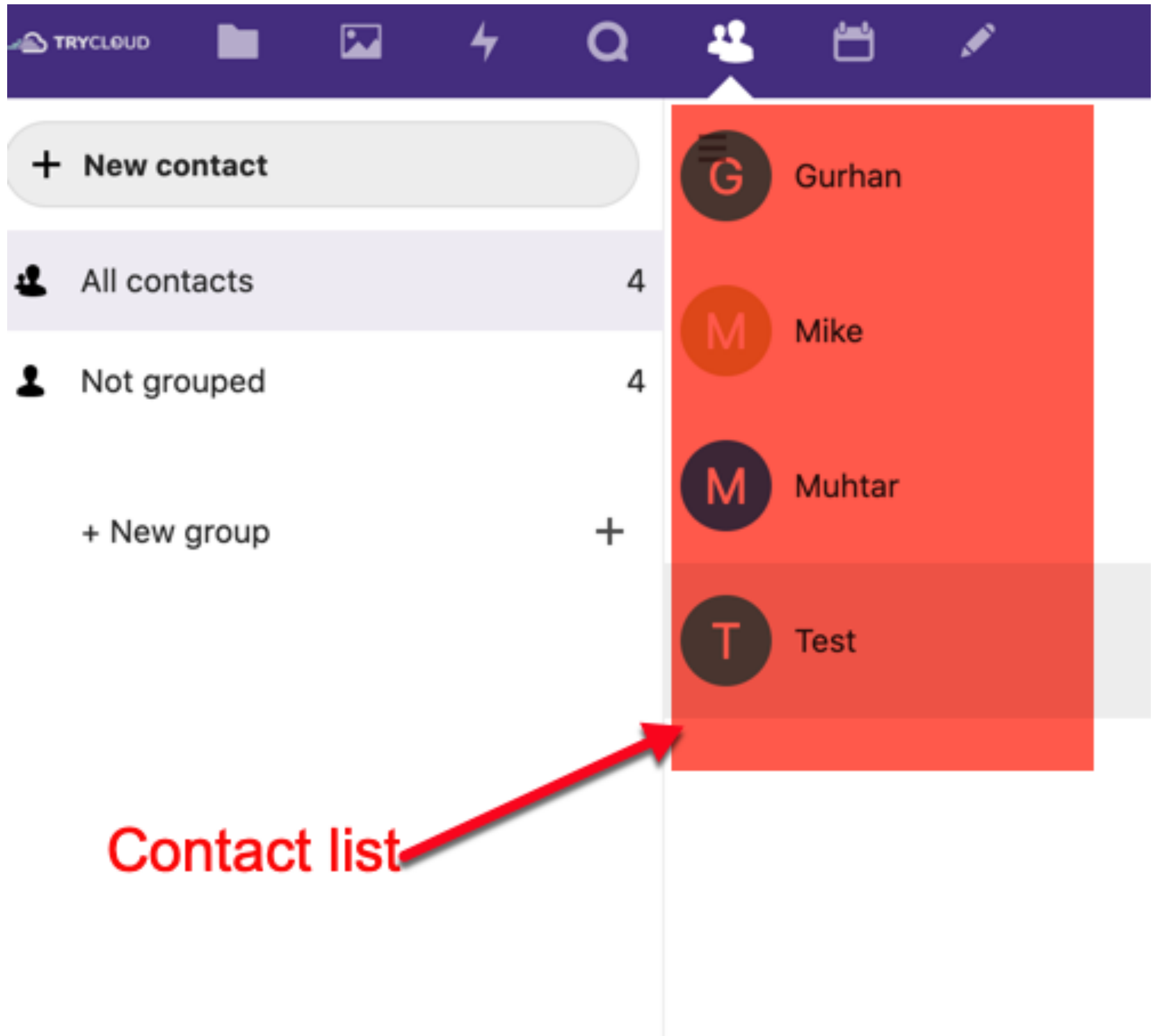


5. **Story**: As a user, I should be able to access to **Contacts** module.

## Test case #3 - verify users can see all the contact names on the contact list

1. Login as a user
2. Click contacts module
3. Verify the contact names are in the list

(Pre-condition: there should be at least 2 contact names are displayed
On the contact list)



Contact list

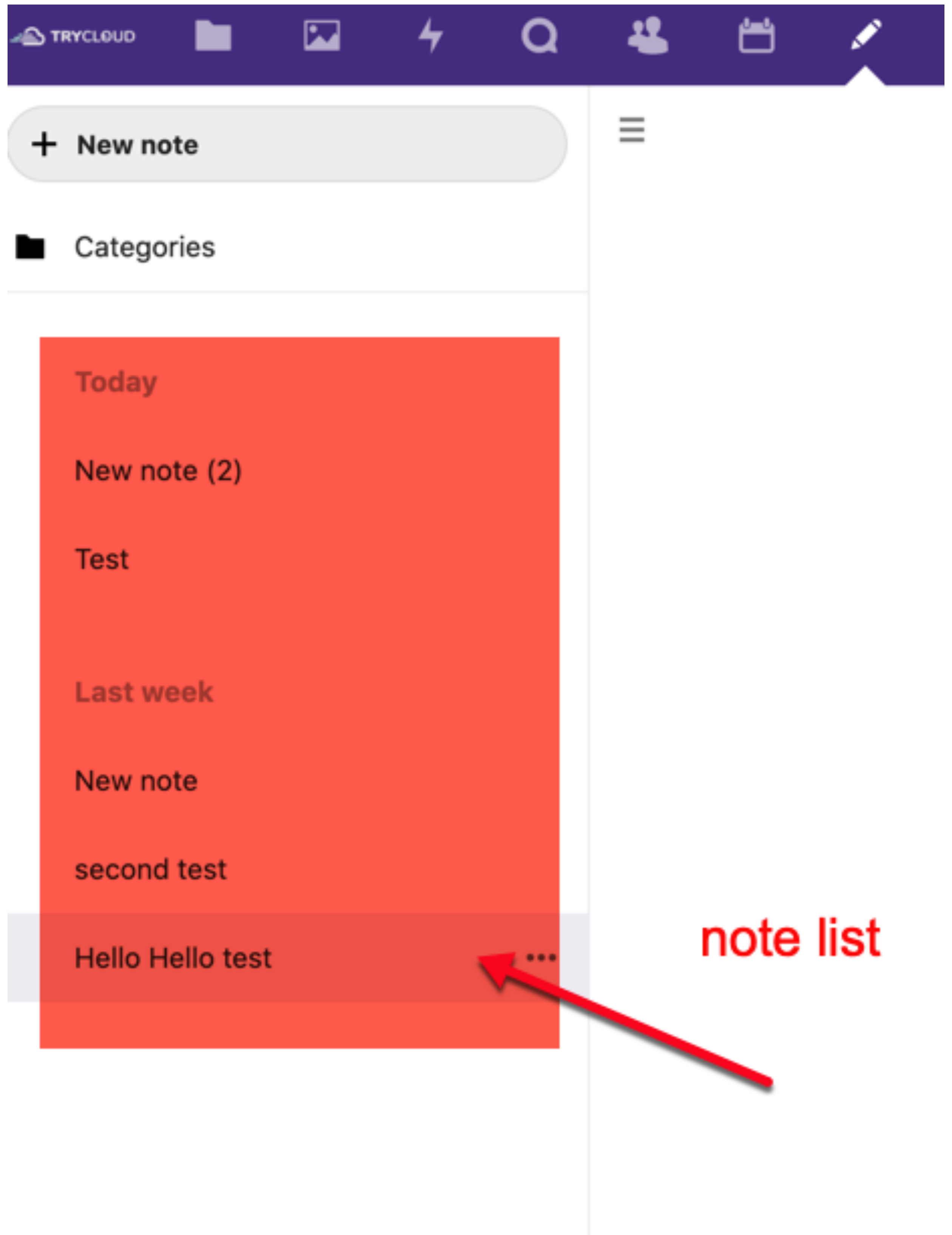
6. **Story**: As a user, I should be able to  access to **Notes** module.

## Test case #1 - verify users can access to Talks module

1. Login as a user
2. Click “Notes” module
3. Verify the page tile/URL is Notes module’s tile

## Test case #2 - verify users can send notes in the page

1. Login as a user
2. Click notes module
3. Click “New Note” button
4. Write a message/Note
5. Verify the note is added on the note list



note list

7. **Story**: As a user, I should be able to access search any item/ users from the homepage.

### Test case #1 - Verify users can search any files/folder/users from the search box.

1. Login as a user
2. Click magnifier icon 🔍 on the right top
3. Search any existing file/folder/user name
4. Verify the app displays the expected result option