Pintailer

User Guide

Version History.

Version	Date	Author	Comment
1.0	12-10-2018	Rinku Sharma	
1.1	25-10-2018	Rinku Sharma	Adding Automation TC and Mapping Related more information
1.2	09-09-2019	Rinku Sharma	Updated based on the changes in the application

In this Tutorial we will learn

1. What is Pintailer?

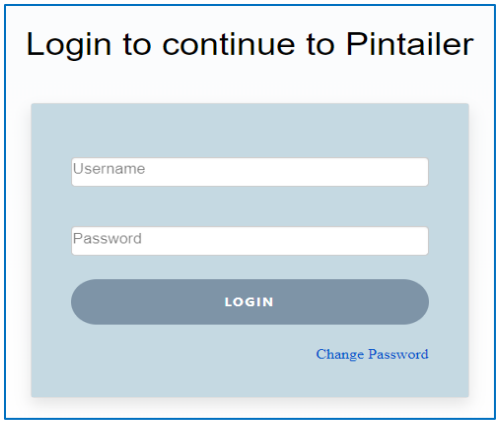
Pintailer – A Tool that helps the QA Teams migrate from Manual Tests to Automation. The name Pintailer is derived from a migrating Bird named Pintail. Another reason that makes us choose the name is that the Tool identifies the actual statistics of Test Automation that any management needs. Before automation testing Boomed up, the industry used the manual testing approach and most of the companies still have their big repositories of Manual test cases. They are willing to move to the world of automation but one thing that keeps them from doing so is that they do not want to lose their Manual Test Cases as they majorly rely on them. Today we have a lot of tools available in market that helps in Automating the Test cases, those which serves as a repository for both manual as well as automated test cases, but what about the retro fit? Where do we link our existing manual test cases with Automation? In the same old manual excel sheets that are tedious and difficult to maintain?

The answer to all the questions and dilemmas is a tool that helps you keep the manual test cases and an equivalent automation test case - Pintailer. This would help one track which test cases are automated and which are pending as the tool ensures one to one mapping between the manual and the automated test cases. Using the spectacular reports the management can visualize the actual status of the work that is done while migrating from manual to automation. The tool provides a reporting structure that is easy to use and the Test Management can rely on it. So what are we waiting for? Let's start using this tool to achieve contemporary and successful Test Management.

In this tutorial we will study how to use Pintailer.

2. Login to Pintailer

To login to Pintailer follow the given steps:
Step 1: Enter a valid "Username" in the Username field.
Step 2: Enter a valid Password using the Password field.
Step 3: Click on "Login" button.



Upon successful login the user will be on the Profile page where they can view their profile.

Note: In case you do not have the Username and Password, then please log a support ticket from Pintailer Website.

3. User Roles

The Users of Pintailer can be assigned with 3 different roles. These roles will have accent to different set of sections within Pintailer.

1. **Admin** – Has complete access to view Setup, Reports and Test Definitions and edit Setup and Test Definition sections
2. **Software Testing** – Has access to view/edit Test Definitions section and View Reports section.
3. **Software Development** – Has view only access to the Test Definitions and Reports section.

4. Project Setup

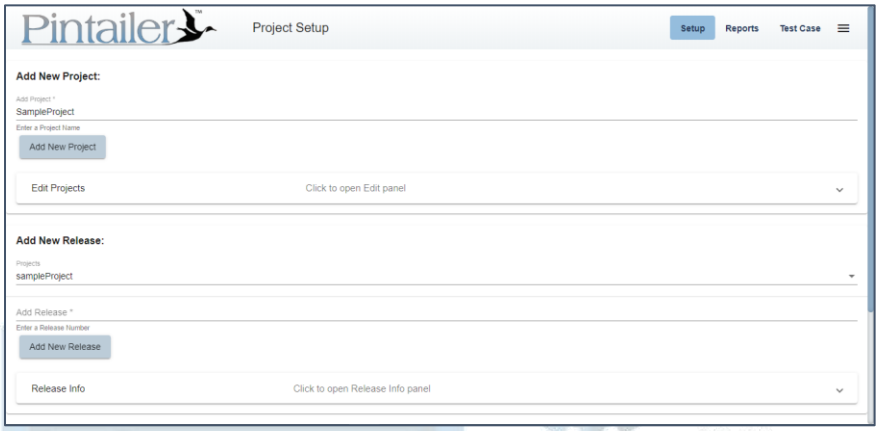
The first thing that is required to start with Pintailer is the Project setup. All the data and reports can be added and viewed only based on the projects created. Using the Project setup page you can set the configurations for the Projects, environment and releases. The Project Set-up is completely role based. The Users would need Admin rights to access this page. So before proceeding make sure that your user has Admin role and on the first login make sure that you are associated with the correct organization.

4.1. Adding and Updating Projects

4.1.1. Adding New Project

To Create Projects, click on Project setup present under Setup in the menu. Once the page is loaded follow the steps:

Step 1: Select the Organization (if you are assigned to more than one organization)
Step 2: Enter the "Project Name" in the Add Project text box
Step 3: Click on Add Project button

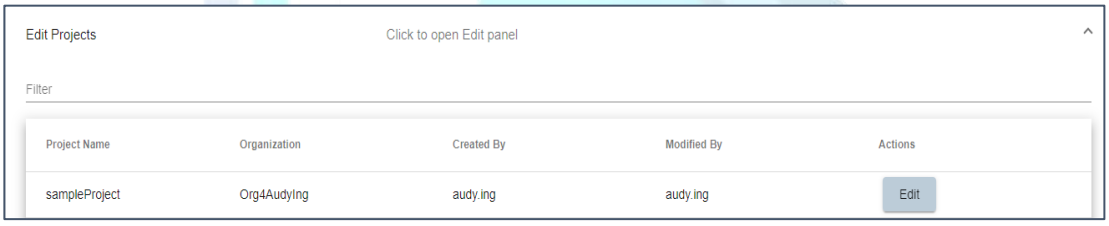


The New Project will be created and a message will be displayed.

Note: To Add a project the user should have the capability of Setup section.

4.1.2. Viewing Existing Projects

To view an existing project, click on the Edit panel present under "Add New Project:" Upon clicking, the panel will expand and the list of all the available project will be displayed with details like - Project Name, Organization, Created By and Modified By.



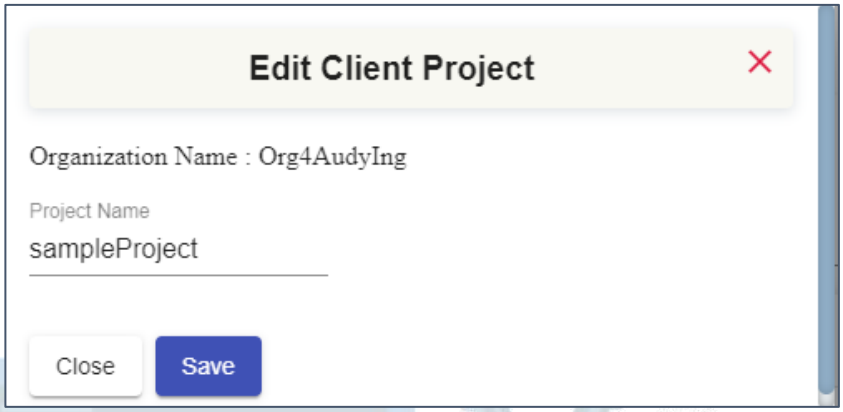
4.1.3. Editing an Existing Project

To Edit an existing project, click on the Edit panel present under "Add New Project:" Upon clicking, the panel will expand and the list of all the available project will be displayed. Follow the steps to edit the project

Step 1: Against each project you will find an Edit button, click on Edit button
Step 2: Update the Project name

Step 3: Click on "Save" button

The updated name will be saved and a message will be displayed.

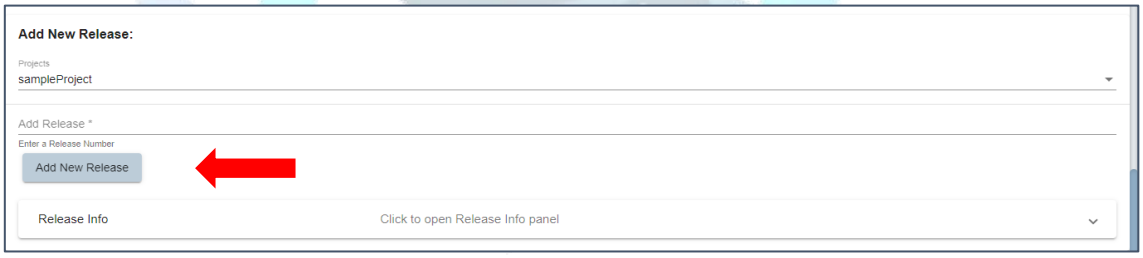


4.2. Adding and Updating Releases

4.2.1. Adding New Release

To create a new Release, click on Project setup present under Setup in the menu. Once the page is loaded follow the steps:

Step 1: Select a Project under which you wish to add a Release
Step 2: In the "Add Release" field enter the "Release Name"
Step 3: Click on Add New Release button

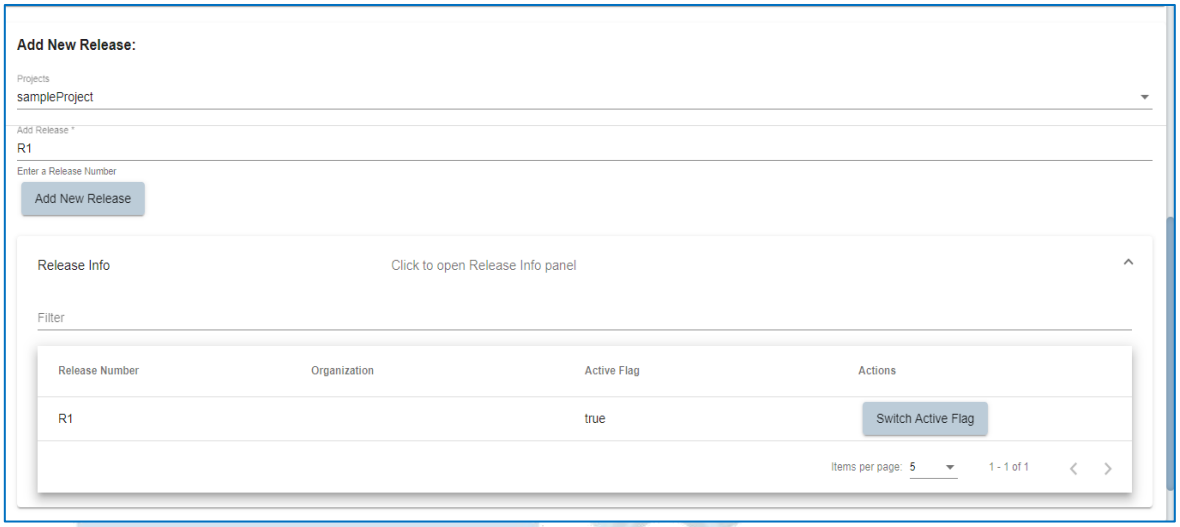


Upon clicking on the button the New Release will be saved and a message will be displayed.

Note: To add a release the user should have the capability of Setup section.

4.2.2. Viewing Existing Release

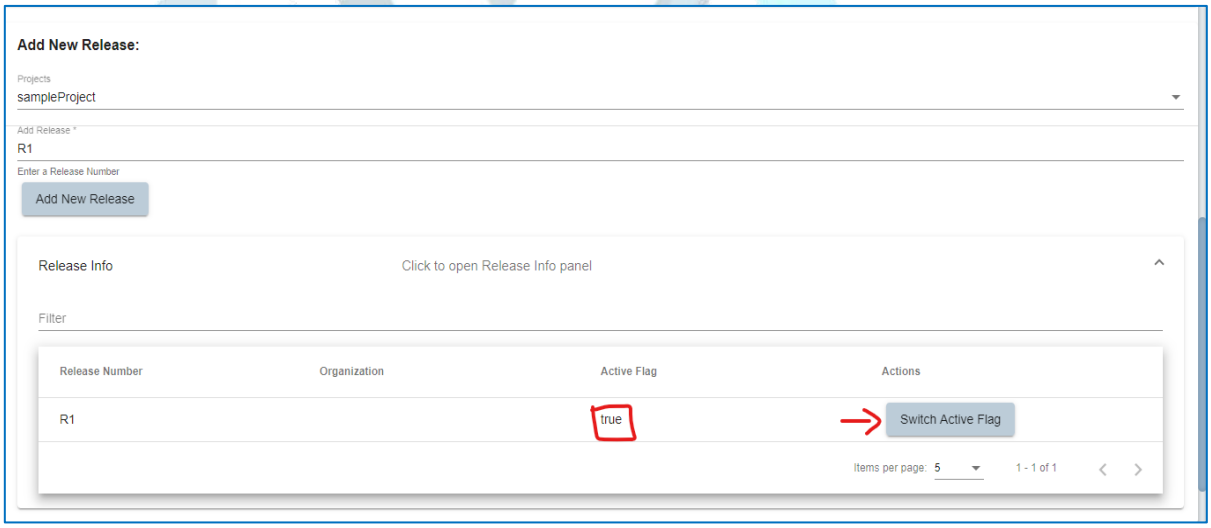
To view an existing release, click on the Release Info panel present under "Add New Release:" Upon clicking, the panel will expand and the list of all the available releases will be displayed with details like – Release Number, Organization and Active Flag.



4.2.3. Switching Active Flag for Existing Release

To switch the flag of an existing release, click on the Release Info panel present under "Add New Release:" Upon clicking, the panel will expand and the list of all the available releases will be displayed.

Step 1: Click on the "Switch Active Flag" button
Step 2: Click on "Ok" button in the pop-up



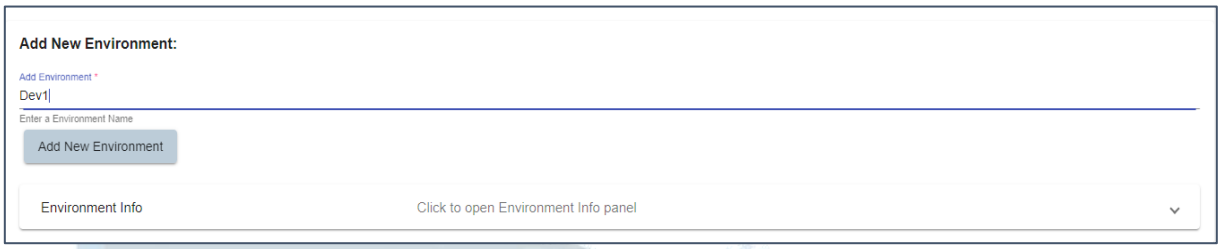
Active flag will be changed and a message will be displayed. By default the release active flag will be "true".

4.3. Adding and Updating Environments

4.3.1. Adding New Environment

To create a new Environment, click on Project setup present under Setup in the menu. Once the page is loaded follow the steps:

Step 1: Select the Organization (if you are assigned to more than one organization)
Step 2: In the "Add Environment" field enter the "Environment Name"
Step 3: Click on Add New Environment button

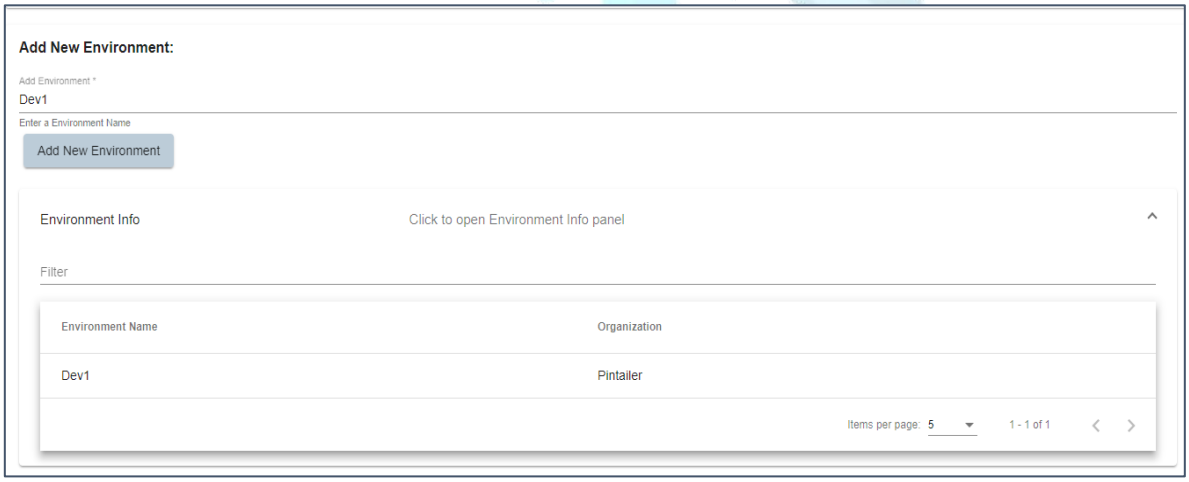


Upon clicking on the button the New Environment will be saved and a message will be displayed.

Note: To Add an Environment the user should have the capability of Setup section.

4.3.2. Viewing Existing Environment

To view an existing Environment, click on the Environment Info panel present under "Add New Environment:" Upon clicking, the panel will expand and the list of all the available Environments will be displayed with details like – Environment Name and Organization.

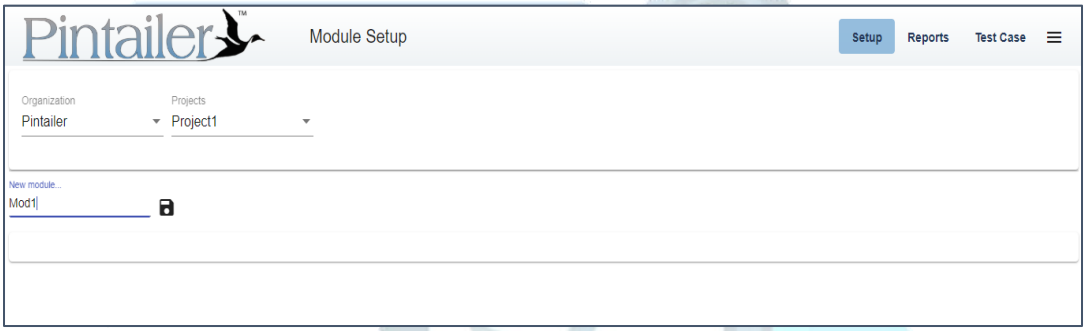


5. Module Setup

Once the Project is created, the modules needs to be added so that later the Test Cases can be created and associated with the relevant modules. To add the modules follow the steps:

Step 1: To add a new Module, click on Module setup present under Setup in the menu
Step 2: Select Project
Step 3: In the New Module field enter a Module Name
Step 4: Click on the save button next to the field.

The New Module will be added and a message will be displayed.



5.1. Creating Sub Modules

The Sub Modules can be created using the following steps under a Parent module:

Step 1: Click on the "+" icon present next to any Module or Sub Module
Step 2: Click anywhere in the UI and the newly added Sub-modules will be saved.

Note: Any number of sub modules can be added in the hierarchy.

6. Test Cases

There are two ways in which a new test case can be added into Pintailer.

1. By creating a new Test Case using the "Add Definition"
2. By Importing Definitions

6.1. Understanding Test Case Template

Before creating or importing Test Cases, let's understand the Fields present in the Definitions as they would be used though out the tutorial from now on.

Field Name	Purpose
TC ID Ref.	A Test Case reference number that would be used to uniquely identify the Test Case.
Module Name	Module added in section 4. This will be used to place the Test Cases under Tree Structure.
Functionality	Sub-module added in section 4.1. This will be used to place the Test Cases under Tree Structure. This would be considered the intermediate module in the hierarchy.
Sub Functionality	Sub-module added in section 4.1. This will be used to place the Test Cases under Tree Structure. This would be considered the lowest module in the hierarchy.
Test Summary	Test Summary refers to the objective of the test case. A test summary should be always be unique
Pre-condition	Pre-condition refers to any specific steps that needs to be fulfilled before the test case is executed
Execution Steps	Step-by-step procedure to execute the test
Expected Result	The expected result of the test
Actual Result	The actual result of the test; to be filled after executing the test
Tester	The name of the author of the test case.
Execution Date	The date of execution of the test
Test Results	Pass or Fail. Other statuses can be 'Not Executed' if testing is not performed and 'Blocked' if testing is blocked.
Test Data	The test data, or links to the test data, that are to be used while conducting the test
Linked Defect	The Defect Number needs to be provided here which is being logged when the Test Case Fails.
Environment	The Test Environment on which the Test Case was executed
Criticality	The priority of the Test Case
IsAutomatable	Indicates that the Test Case is Automatable
Remarks	Any comments on the test case or test execution.

File Name	Name of the file where the relevant automation exists
Test Case No.	Test Case Scenario or method name for the Test Case in the automation file
Applicable	Indicates whether the Test Case is currently Active or Inactive.

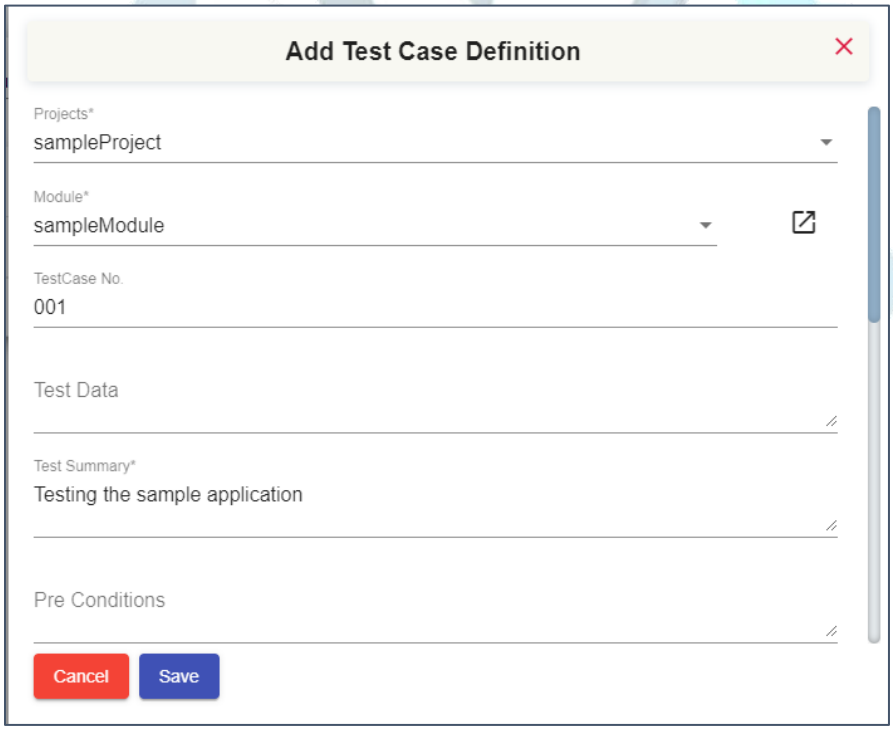
PS: In case you use a template that is extremely different from the one provided by us the please contact us and we can customize the tool as per your preference.

6.2. Creating New Test Cases

To create new Test Cases follow the steps:

Step 1: Click on Definitions present under Test Case from the Menu
Step 2: Click on "Add Definition" button
Step 3: Select Organization, Project and Module
Step 4: Enter a Test Case Number, Test Summary, Pre-condition, Tag (Priority), Execution Step and Expected Result
Step 5: Click on save button

The New Test Case will be created and saved into the repository.



6.3. Importing New Test Cases

6.3.1. Prerequisites while Importing Test Cases

There are some basic information or constrains that needs to be taken in consideration before we import any new test case in Pintailer:

- If the value of Automatable is marked 'yes', Automated is 'yes' and a "Test Case No." is provided then the test case mapping will be done automatically For TestNG.
- While mentioning the Automation File Name the name format should contain "_" (an underscore). Example: file_name.csv

6.3.2. Importing New Manual Test Cases

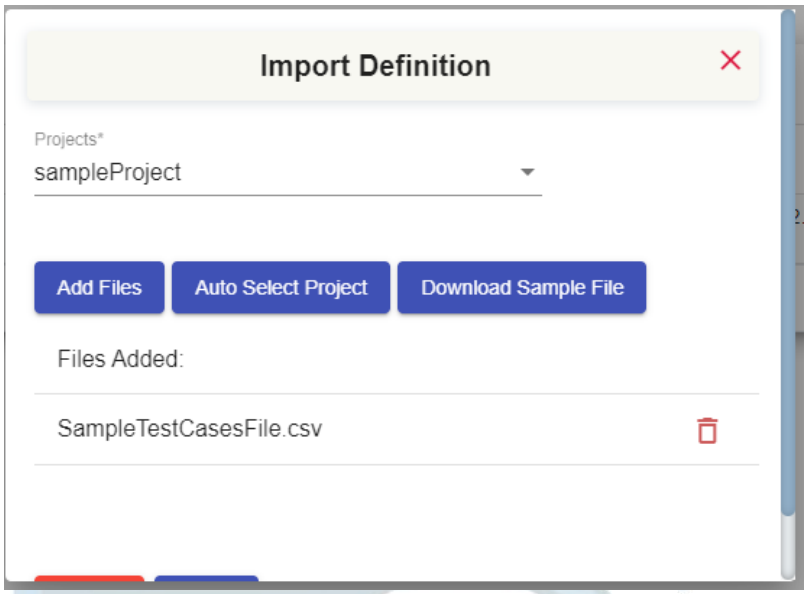
To Import Test Cases in Bulk follow the given steps:

Step 1: Click on Definitions present under Test Case from the Menu
Step 2: Click on "Import Definition" button
Step 3: Download the Sample File
Step 4: Add all the Test Cases in the file with the Columns filled in as given in Step 5
Step 5: Enter Module, Functionality, Sub Functionality, Test Case Number, Test Summary, Pre-condition, Tag (Priority), Execution Step and Expected Result
Step 6: Save the CSV File in your system with the Name format as <Project>_<Filename>.csv
Step 7: In the "Import Definition" pop-up Select Organization and Project
Step 8: Click on "Add Files"
Step 9: Select the Saved CSV file
Step 10: Click on Upload

The New Test Case will be imported and saved into the repository.

Note: In case there are any discrepancies in the import file all those would listed out by the system and the file will be uploaded only once

those are rectified. In case of Duplicate Test Cases, the test cases will not be uploaded



6.3.3. Importing Existing Manual Test Cases with Automation

To Import test cases that have Automation already done follow the below steps:

Step 1: Click on Definitions present under Test Case from the Menu
Step 2: Click on "Import Definition" button
Step 3: Download the Sample File
Step 4: Add all the Test Cases in the file with the Columns filled in as given in Step 5
Step 5: Enter Module, Functionality, Sub Functionality, Test Case Number, Test Summary, Pre-condition, Tag (Priority), Execution Step, Expected Result, IsAutomatable, IsAutomated, File Name and Test Case No.
Step 6: Save the CSV File in your system with the Name format as <Project>_<Filename>.csv
Step 7: In the "Import Definition" pop-up select Project
Step 8: Click on "Add Files"
Step 9: Select the Saved CSV file
Step 10: Click on Upload

The New Test Case will be imported and saved into the repository.

Automation in TestNG

Here, in case the Test Cases are automated in TestNG then these fields would mark/map the automation for those Test Cases automatically by

just providing Filename, Test Case Number and Automatable/Automated Yes. In Case the automation for the Test Cases would be done in future then these fields can be left blank and would need to be updated manually in Test Case Edit.

Automation in Selenium (Using Cucumber)
If the Automation is done using selenium, then only mentioning the name of the Cucumber feature file in the File Name column and leave the next column "Test CaseNo." Blank.

Note: In case there are any discrepancies in the import file all those would listed out by the system and the file will be uploaded only once those are rectified. In case of Duplicate Test Cases, the test cases will not be uploaded

6.4. Editing Existing Test Cases

The existing test cases can only be edited in the Pintailer's Edit functionality (User will not be able to Import the edited test cases). Follow the steps to edit the existing Test Cases

Step 1: Click on Definitions present under Test Case from the Menu
Step 2: Search the existing Test Case.
Step 3: Right click on the test case and then click on Edit button
Step 4: Update the fields that needs to be changed.
Step 5: Click on Save button

The Test Case will be updated and saved into the repository.

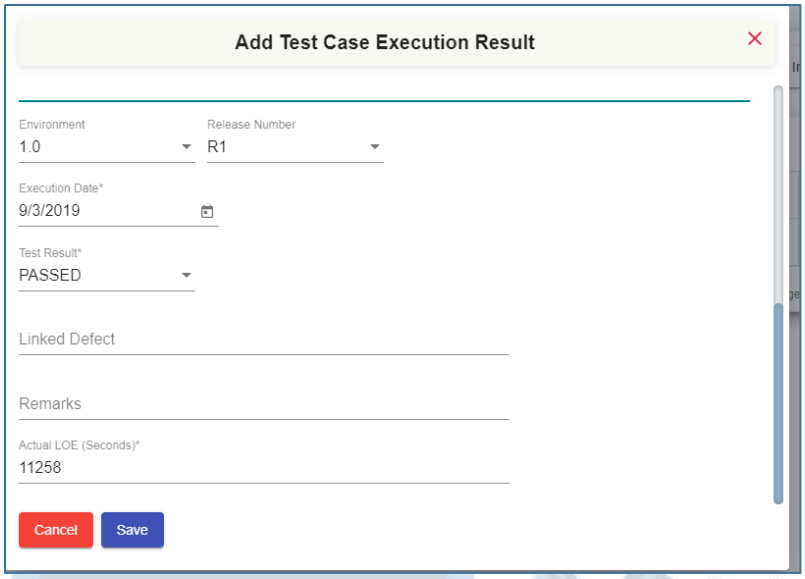
6.5. Updating Manual Execution Results

The existing test cases can be executed manually by the executioner and the results can be updated using the following steps:

Step 1: Click on Definitions present under Test Case from the Menu
Step 2: Search the existing Test Case.
Step 3: Right click on the test case and click on Add Execution Results
Step 4: Update the "Add TestCase Execution Result" section with the information like Environment, Release Number, Execution Date, Test Result, Linked Defect, Remarks and Time taken to execute test cases.
Step 5: Click on Save button

The Test Case Results will be updated and saved into the repository.

PS: Make sure that the Release mapping is already done before you update the results.



6.6. Importing Automated Test Files

In Pintailer you can add 3 types of automated test cases

1. Manual Test Cases Automated with TestNG Implementation
   Test Cases that are automated using the TestNG framework
2. Manual Test Cases that are automated using Cucumber
   Test Cases that are automated using the cucumber framework
3. 100% Automated Test Cases using Cucumber
   Another major feature that Pintailer provides is the Automatic Mapping of Automation Test Cases. Many times we observe that there are cases that the Testing is accomplished completely using Automation. In this case there are no counts, scenarios etc. Pintailer provides the user flexibility to just upload their Feature file into the system and a corresponding manual test definition would be created for all the scenario present in the feature file. To utilize this feature please refer the below sections.

There are multiple things that needs to be taken care before the Automated Test Cases are imported into the Pintailer system

6.6.1. Prerequisites while Importing Automation Test Files

Before importing a feature/xml file into the Pintailer system there are some basic rules that needs to be kept in mind.

For TestNG implementation:

- In case of a TestNG Script, to mark the presence of a Test Case, there should be an annotation like "@Test(description =

"ProjectName_JavaFileName :: TestCaseSummary")" which is an annotation mandated by TestNG. This annotation is used by Pintailer to identify the Test Cases present in the script. With this annotation we create the methods for each Test Case. These methods will be considered as the Test Cases

- Example:

  @Test(description = "LoginPageTest :: TC_00:Login Validation - Wait for login page to Load Successfully", groups = "P1")

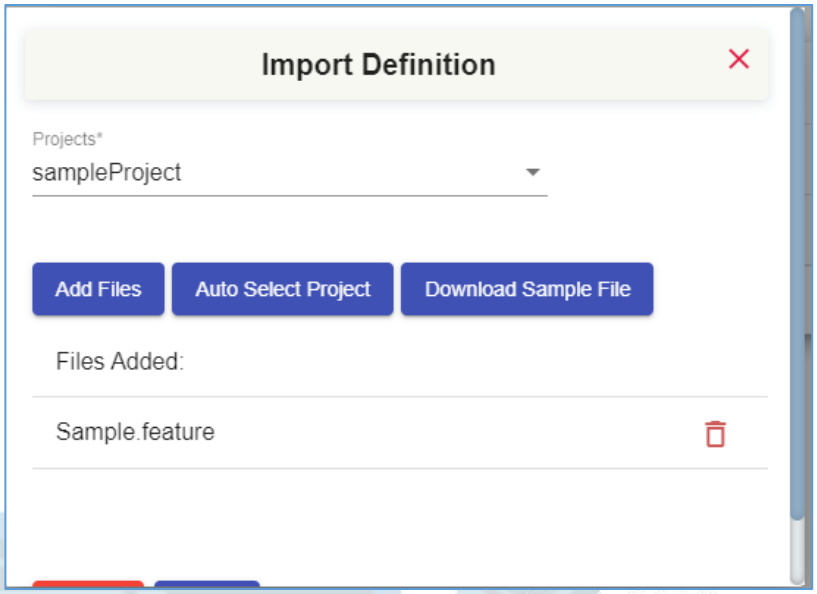
For 100% Automation wit cucumber implementation

- In Case of Feature File there should be a comment with the Test Case number and a small description. This way the steps related to the Test Case will be taken as Execution Steps for that test case
- If a statement starts with "##" followed by 'TC<##>: <Description>' it will be considered as a test case.
- Line with '~' will be ignored completely in comments
- If a statement starts with a Keyword then it will be ignored (Note: If any organization wants to set certain words as keywords then please contact the Pintailer support Team.).
- Lines with the keywords Given, When, Then will be inserted in database but will be disabled and will not be displayed on ui untill it is enabled

6.6.2. Importing Automation Test Files

To Import Feature files for Automation mapping, follow the given steps:

Step 1: Click on Definitions present under Test Case from the Menu
Step 2: Click on "Import Definition" button
Step 3: Add all the files of .feature or .xml format
Step 4: In the "Import Definition" pop-up Select Organization and Project
Step 5: Click on "Add Files"
Step 6: Select the feature file
Step 7: Click on Upload



Note: In case there are any discrepancies in the import file all those would listed out by the system and the file will be uploaded only once those are rectified.

6.7. Importing Test Execution Results

Before importing any file keep these basic rules in mind:

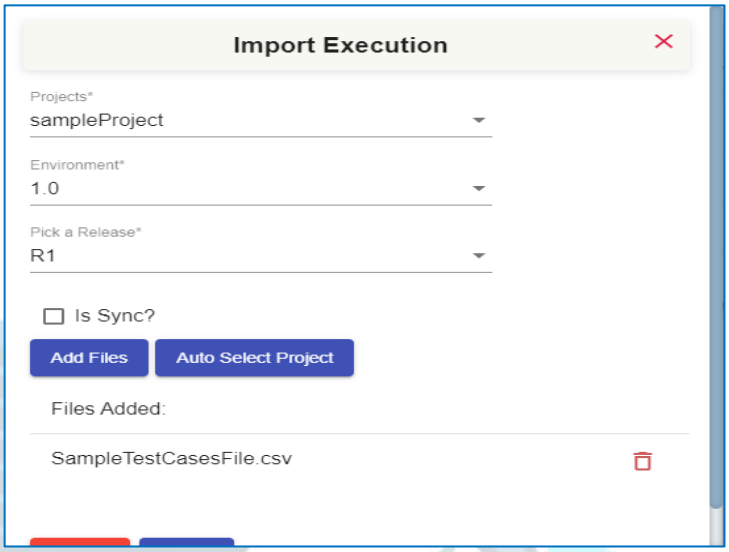
- Date must be in "YYYY-MM-DD HH:MM:SS" format, else import will not be done and discrepancies would be listed out.
- Duration must be provided in seconds
- Test step definition must be provided without keywords
- Test case and Release Id mapping must have been done correctly for getting the correct counts in the Execution Reports. i.e. json, xml and csv will be imported but data will be visible only when the test cases are mapped with release id
- For TestNg - File name with "_" should be provided in import CSV file at the beginning of the description as <FileName> :: <Rest Of the description>

6.7.1. Importing Manual Test Results

To Import manual Test Results follow the given steps:

Step 1: Click on Definitions present under Test Case from the Menu
Step 2: Click on “Import Execution” button
Step 3: In the “Import Execution” pop-up Select Organization, Project, Environment and Release Number
Step 4: Click on “Add Files”

Step 5: Select the CSV file (Note: The file should be in the format it was uploaded in the tool or it should have been downloaded from the reports section and the Fields corresponding to Execution should be filled)
Step 6: Click on Upload

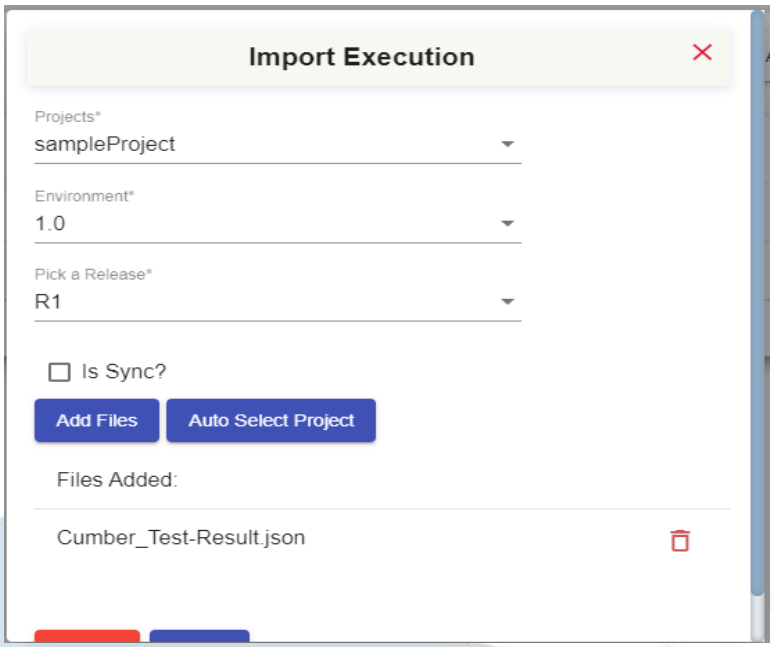


Note: In case there are any discrepancies in the import file all those would listed out by the system and the file will be uploaded only once those are rectified.

6.7.2. Importing Automated Test Results

To Import Json/XML files for Automation mapping, follow the given steps:

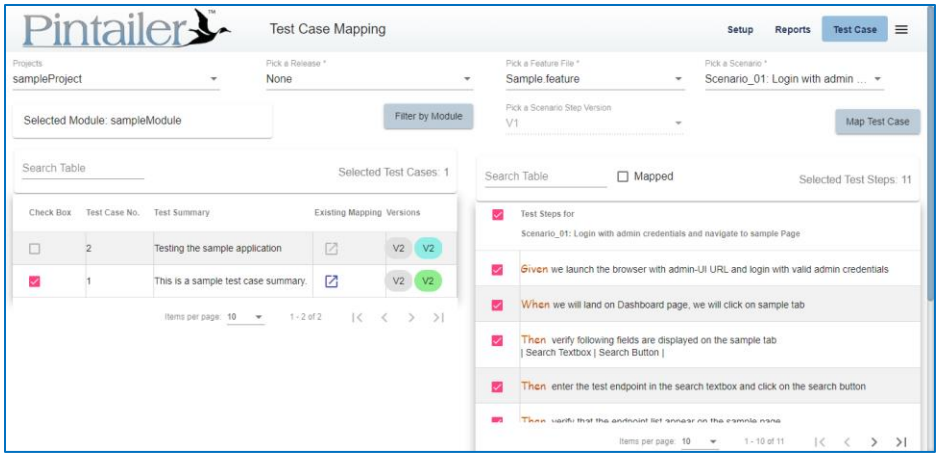
Step 1: Click on Definitions present under Test Case from the Menu
Step 2: Click on "Import Execution" button
Step 3: Add all the files of .json or .xml format that are applicable
Step 4: In the "Import Execution" pop-up Select Organization, Project, Environment and Release Number
Step 5: Click on "Add Files"
Step 6: Click on Upload



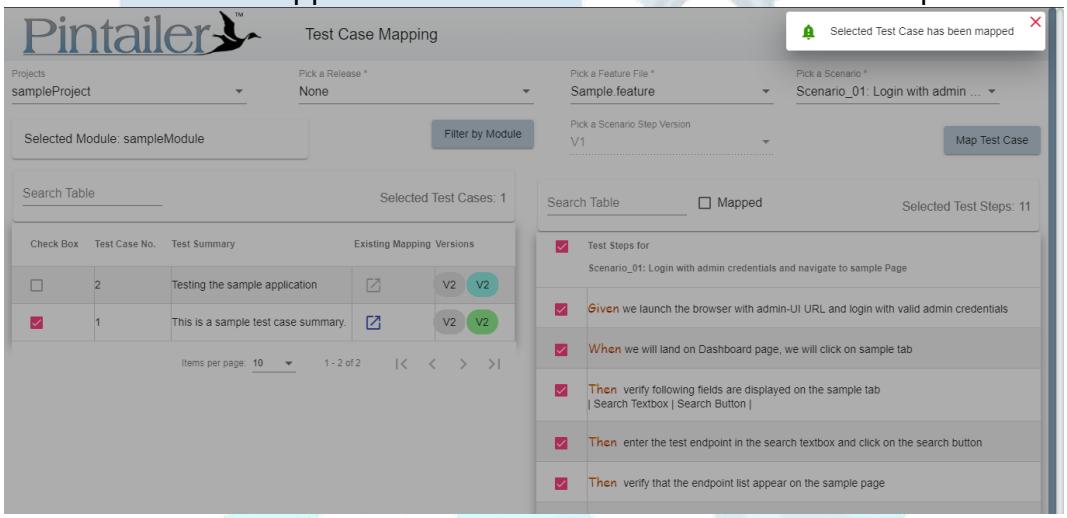
7. Test Automation Mapping

Pintailer's core feature is the mapping of Automated Test Cases with their Manual Test Cases. To map the existing manual test cases with their respective automation script, follow the below steps:

Step 1: Click on Automation Mapping present under Test Case from the Menu
Step 2: Select 'Project'
Step 3: Select the Module for Manual Test Case. All the manual test cases present in the selected module will be listed in the left side of the screen
Step 4: Selecting a Release is optional. It will by default be Null.
Step 5: From the drop downs present in the Right side, select the Feature File and its corresponding scenario (that contains the automation steps)
Step 6: If there exists multiple versions of the feature file then select the version you want. Else the default selection would be the latest version.
Step 7: Select the Test Case from the left side
Step 8: Select the Steps from the right side
Step 9: Click on Map Test Case button.



Test Cases would be mapped and the count will be added in the reports



8. Edit Feature File

To ease the job of a QA Engineer, Pintailer provides a feature to edit the .feature file within Pintailer application. This would simplify the users job and would enable user to use the latest implemented file to make the changes and also would not disrupt the existing test automation mapping. Once the updates to the feature file is done then the user can export it and use it for implementing the automation steps. To achieve this, follow these steps:

8.1. Editing Existing Steps

Step1: Click on Edit feature file present under Test Case in the menu.
Step2: You will be navigated to Edit Feature File page
Step3: Pick a project from the dropdown
Step4: Pick a Feature File from the dropdown

Step5: The Feature File will open in the bottom pane. Note: By default the latest version of the Feature file will be opened.
Step6: If you need to modify a step then, go to the step and directly start modifying it. Once the edit is completed.
Step7: Click on Save button.

The colour of the edited line would be changed to yellow

8.2. Adding new Step in an existing feature file

If you need to add a new Step, then go to the step where a new step is supposed to be added and click on the upward or downward arrow to add a step above or below that particular step. The color of the added step will be changed to green. Save the Feature file once the edit is completed.

8.3. Deleting an Existing Step

If you want to delete a step from the existing feature file then click on the Delete icon present next to the step. The deleted step color will be changed to Red. Save the Feature file once the edit is completed.

8.4. Adding/Deleting New scenario in existing feature file

You can add a completely new scenario in an existing feature file. To do this, click on the + icon present next to any scenario and a new scenario will be added after that scenario.

Similarly, if an entire scenario needs to deleted then click on the delete icon next to the scenario.

8.5. Generating Java Steps

Pintailer provides you with an added feature of generating java steps skeleton for a feature file. To do this click on Generate Java Steps.

If you are fine with the skeleton then you can export the java file by clicking on the "Export Java file" button

8.6. Export Feature file

If you wish you can export any feature file that is present in Pintailer and use it for sending to anyone or to implement the Java test steps. To export the feature file click on "Export feature file"

8.7. Comparing different versions of Feature file

You need not use any third party tool to compare the different versions of feature file. Pintailer provides you with an inbuilt feature that enables you to compare them within Pintailer. To do so follow the steps:

Step1: Pick a Project
Step2: Pick a Feature File
Step3: Select Old Version
Step4: Select New Version
Step5: Click on Compare button

The comparison results will be shown in the bottom grid. The colour coding is as follows

Action	Colour
Step Added	Green
Step Deleted	Red
Step Edited	Green

9. BDD Editor

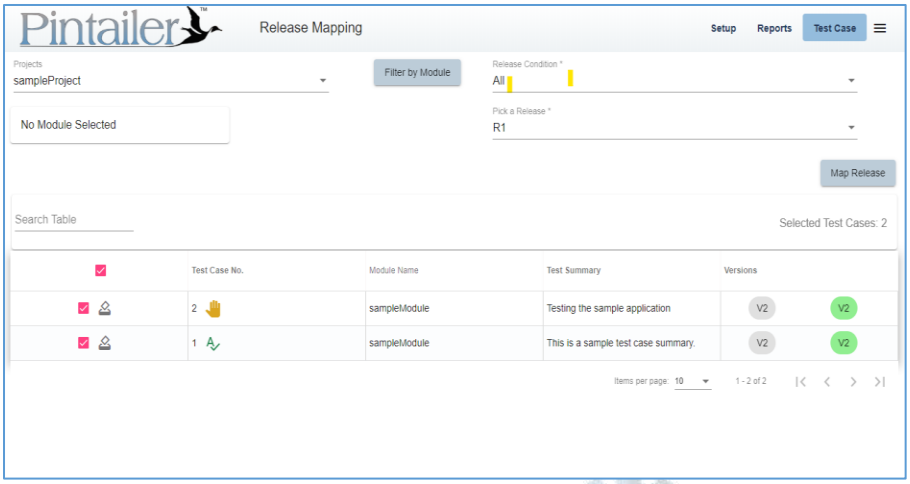
In Pintailer, you can start creating a fresh feature file without worrying about the syntax. Pintailer provides you with a skeleton of syntax. All you have to do is just start by entering the text for the test cases.

Click on the hamburger (3 lines) present at the right corner and then click on BDD Editor. Start updating the feature file and once done you can also create its java skeleton. Once you are ready export the file to use it in your test suite.

10. Test Case Release Mapping

To map the existing test cases to a particular release follow the given steps:

Step 1: Click on Release Mapping present under Test Case from the Menu
Step 2: Select the Release Condition Active/Inactive
Step 3: Select the Release (to which the Test Cases needs to be mapped)
Step 4: Select 'Organization', 'Project' and 'Module'
Step 5: All the Test Cases will be listed in the bottom section
Step 6: Select the Test Cases that needs to be added to the release
Step 7: Click on 'Map Release' button



The Test Cases will be mapped to the Release and a success message will be shown

11. Reports

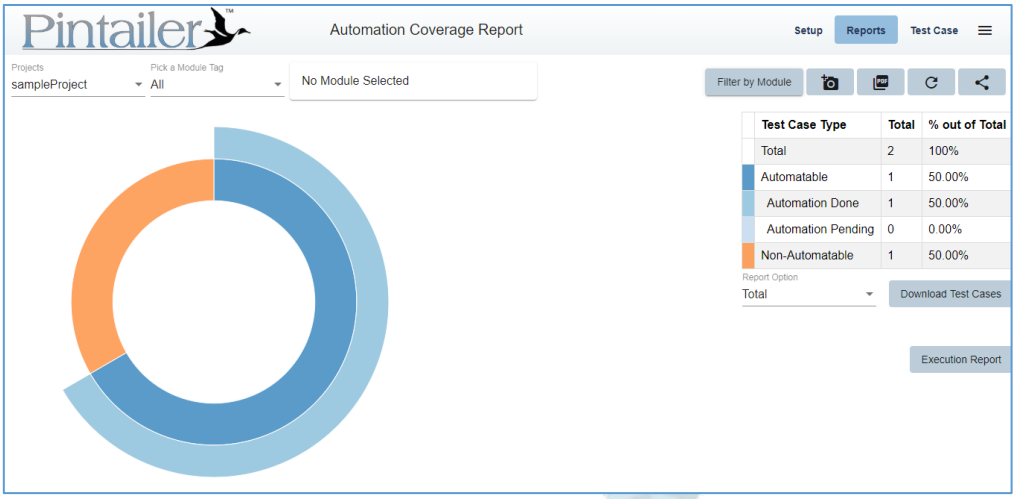
There are 3 types of Reports currently available in Pintailer

1. Test Coverage
2. Test Execution and
3. QA Progress

11.1. Test Coverage Report

The Coverage report gives the exact count of Test Cases with their Automation. The report contains 5 fields and its corresponding Graph. This report is very useful for the management as it gives the exact count of test cases that are automated and the count of test cases that are not-automated with its percentage.

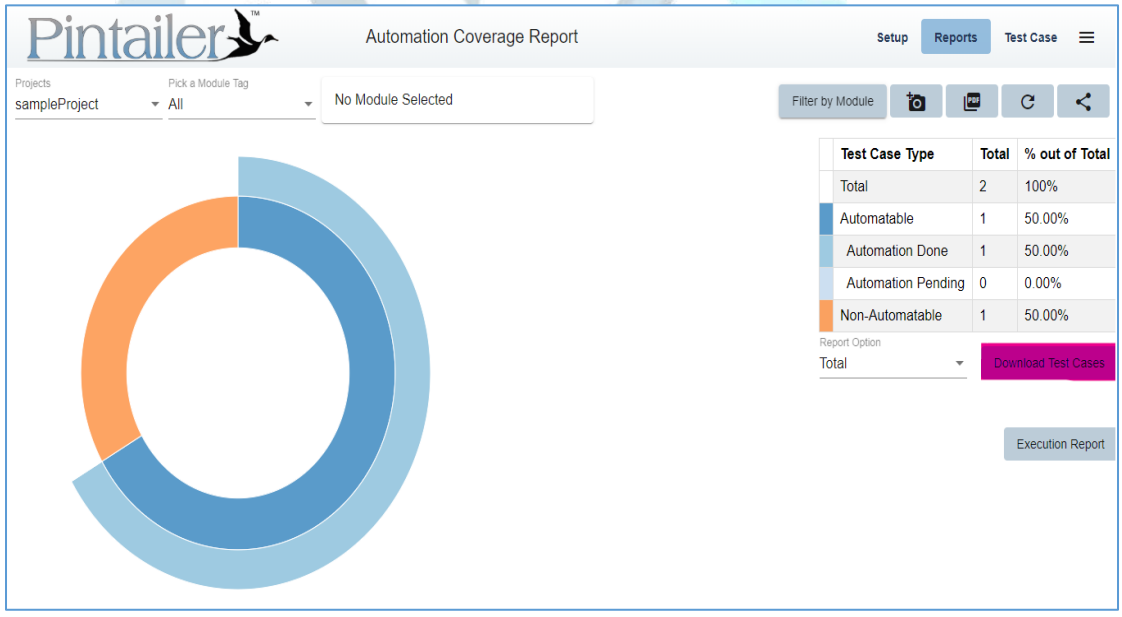
Fields	Definition
Total	Contains the Total number of Test Cases that are added onto the selected module
Automatable	Holds the count of test cases that could possibly be automated
Automation Done	Contains the count of test cases that are automated and its corresponding mapping with manual Test Cases is done in the tool
Automation Pending	Contains the count of test cases that are Automatable but are not yet automated and mapped with its corresponding manual Test Case
Non-Automatable	Contains the count of Test Cases that cannot be automated



11.1.1. Downloading Test Cases from Coverage Report

To download the CSV of the Total, Automated, Non-automated, Automation Done and Automation Pending Test Cases follow the steps:

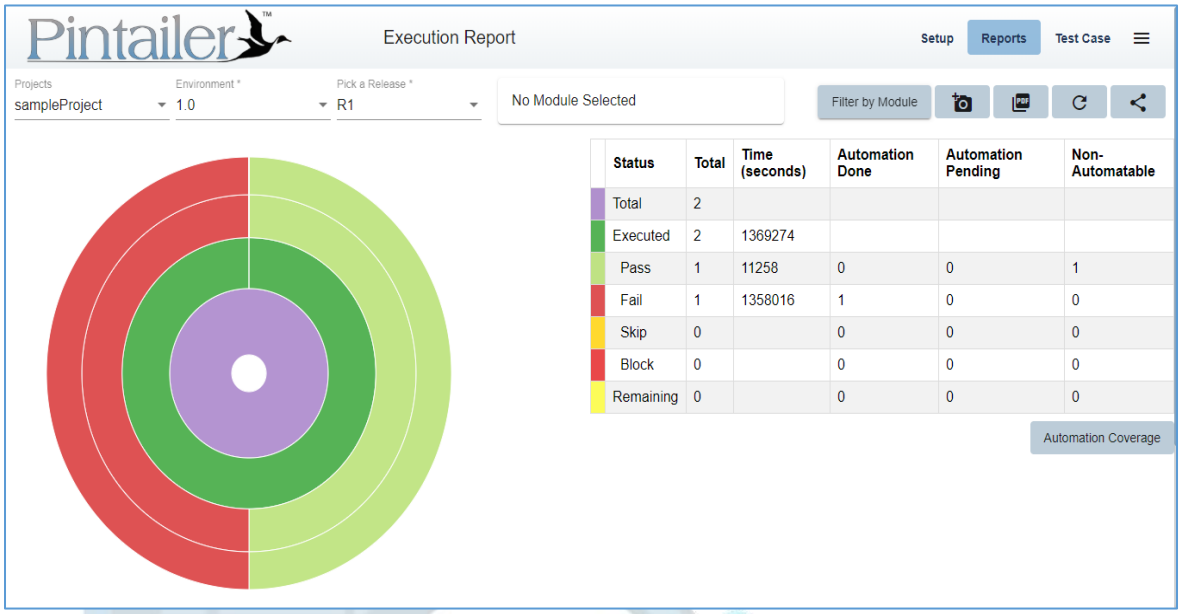
Step 1: In the Coverage Report select the Type Test Cases that needs to be downloaded
Step 2: Enter a Name for the File
Step 3: Click on Download Test Cases button.



11.2. Test Execution Report

The execution report displays the exact count of the Test Cases that are executed and their corresponding status. The Report also displayed

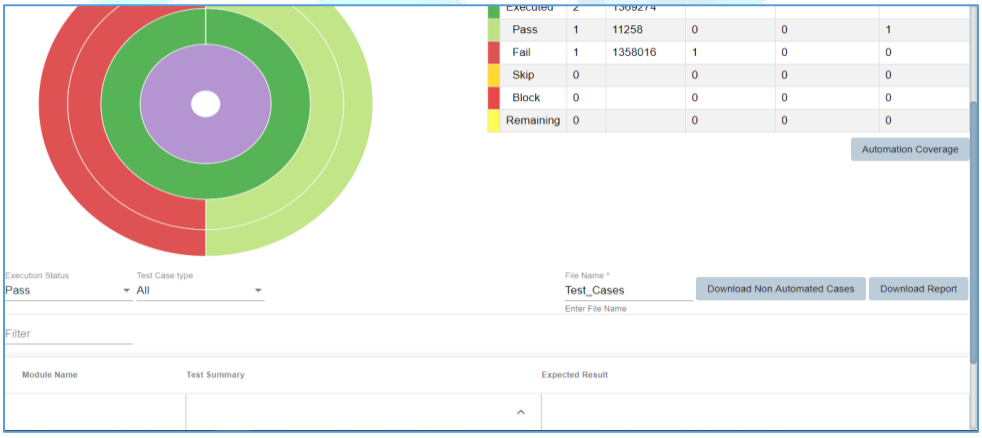
the counts form the coverage report i.e. Under each status how many test cases are automated, non-automated etc



11.2.1. Viewing the Manual Test Cases in the Report

The user has the ability to view the executed test cases in bottom section of the report. To do this user can use any of the following methods:

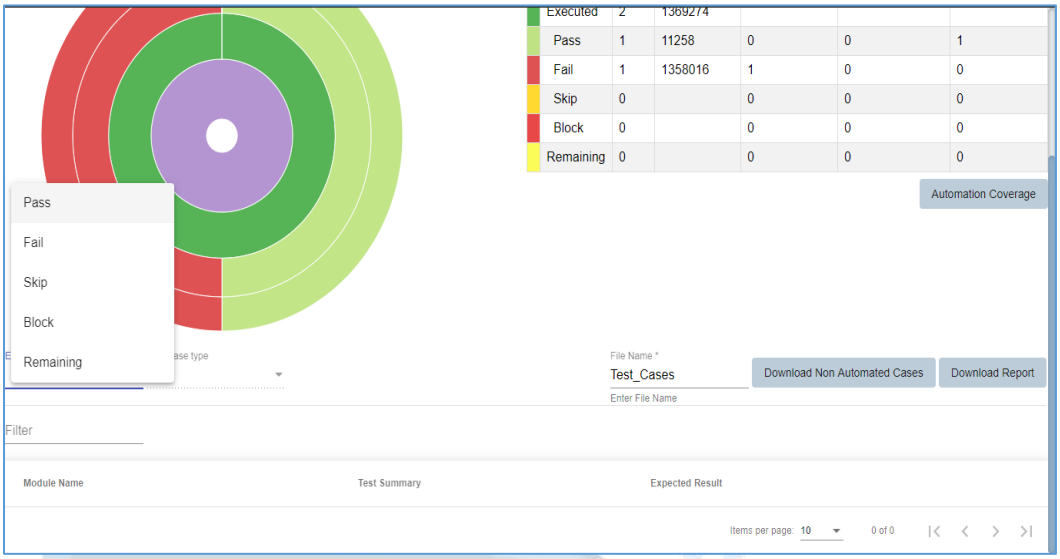
1. By clicking on one of the Execution Results



2. By selecting the Execution Status from the dropdown

For example if a user wishes to see the test cases that are failed then they just needs to Click on the Fail from the table or select Fail form

the Execution status dropdown and the Test cases will be listed in the bottom.



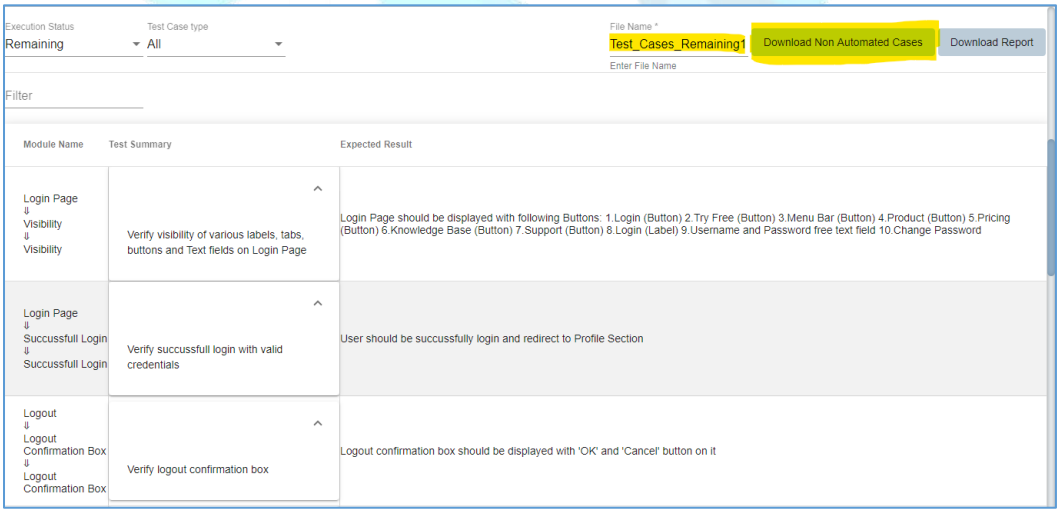
11.2.2. Downloading the Remaining Manual Test Cases

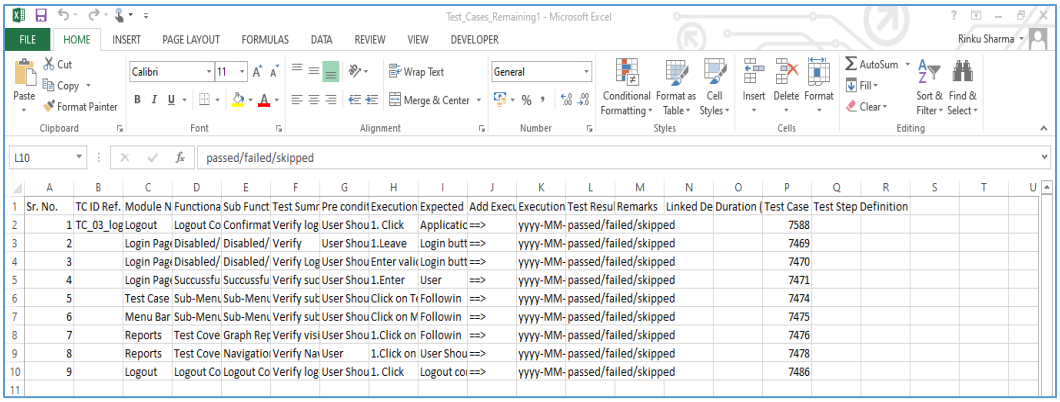
To download the CSV of the remaining Manual Test Cases follow the steps:

Step 1: In the Execution Report select the Type of Execution Status of the Test Case that needs to be downloaded

Step 2: Enter a Name for the File

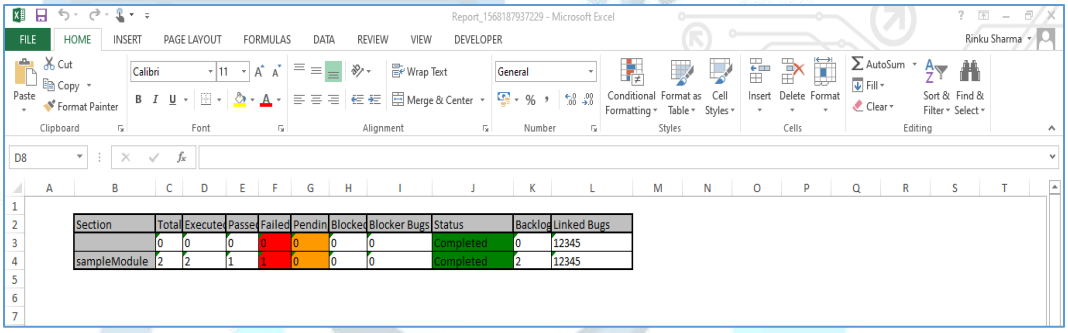
Step 3: Click on Download Non-Automated Test Cases.





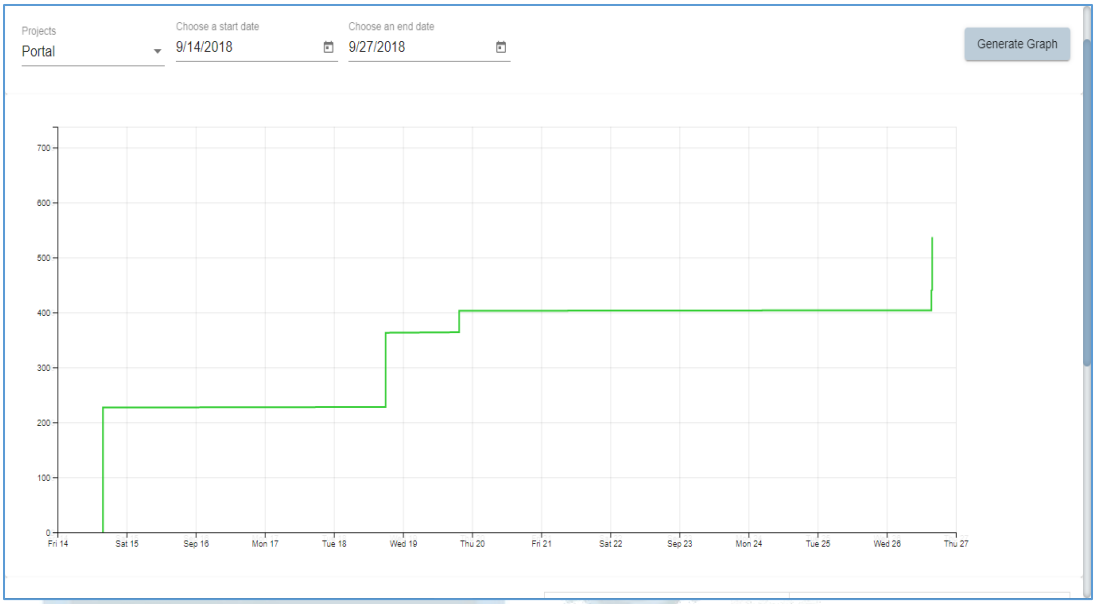
11.2.3. Downloading the Execution Report

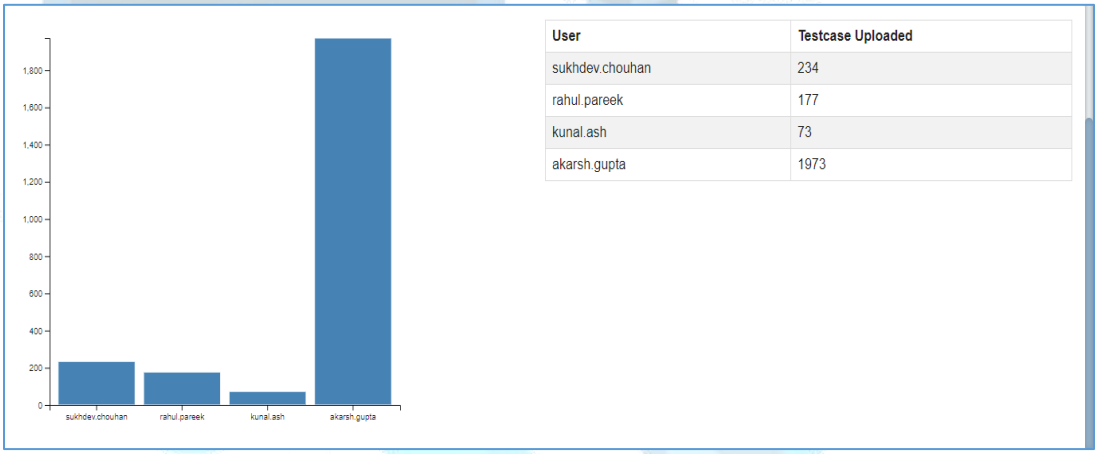
To download the CSV of the report Enter a Name for the report and click on Download Report.



11.3. QA Progress Report

This reports shows the pictorial representation of the number of test cases added for the selected Project during the selected duration. This report also list the number of test cases added by a particular user.





User	Testcase Uploaded
sukhdev.chouhan	234
rahul.pareek	177
kunal.ash	73
akarsh.gupta	1973