**Nx2me Clinician Portal Automation Reference Document**

**Document Number: YYY**

**Revision: A**

**Table of Contents**

[1 Introduction 3](#_Toc63936751)

[2 Test Environment 3](#_Toc63936752)

[2.1 Test Setup 3](#_Toc63936753)

[3 Nx2me Clinician Portal UI 5](#_Toc63936754)

[4 User Interface Verification Test Case 7](#_Toc63936755)

[5 Functionality Verification Test Case 9](#_Toc63936756)

[6 Flowsheet Statistics Verification Test Case 12](#_Toc63936757)

[12](#_Toc63936758)

[7 Result Verification 13](#_Toc63936759)

[8 Source Code 14](#_Toc63936760)

[8.1 Source Code Structure 14](#_Toc63936761)

[8.1.1 Java and Selenium 14](#_Toc63936762)

[8.1.2 User Interface Application 20](#_Toc63936763)

[8.2 Updating Code Guidelines 20](#_Toc63936764)

[8.3 Reference 21](#_Toc63936765)

[9 Environmental Issues and Solutions 21](#_Toc63936767)

[10 Script Update Policy 21](#_Toc63936768)

[11 Revision History 21](#_Toc63936769)

# Introduction

The purpose of this document is to describe about how to use test automation script for verifying Nx2me Clinician Portal web application using Google Chrome and Microsoft Edge browsers whenever the browser is updated.

# Test Environment

The purpose of this section is to describe the test environment for establishing the setup in the test PC, for launching the test automation scripts:

1. Ensure that Windows 10 is installed in the test PC.
2. Ensure Java (minimum 14.0.1 or above) is installed in the test pc.
   1. Open command prompt and type “java -version” and click enter button
   2. Displayed Java version should 14.0.1 or above
   3. If the displayed Java version is lesser than 14.0.1, then please use the below link to upgrade Java to the latest version:

<https://www.oracle.com/java/technologies/javase-jdk15-downloads.html>

1. Ensure that Microsoft Office with Excel is installed in the test PC.
2. Ensure that Google Chrome, Microsoft Edge browsers are installed in the test PC.
3. Ensure that the test PC is connected with VPN.
4. Ensure that “Connectivity.TestToolGUI “is installed in the test PC in the root drive **(C:\)**.
5. Ensure that “ NxStorage Credentials is provided to launch the Connectivity.TestToolGUI if the test PC is outside the domain..
6. Download the “**ClinicianPortal\_Automation\_TP2186**.zip” attached to **TP2384** item in **Agile.**
7. Extract **‘ClinicianPortal\_Automation\_TP2186.zip’** to root drive **(C:\)** in the Test PC**.**
8. Ensure that the folder named “**ClinicianPortal\_Automation\_TP2186**” is created in the Test PC in the root drive.
9. Run the “**DllReg.bat**” file as Administrator to register the “**AutoItX3\_x64.dll**” in the test PC.
10. Navigate to the “**ClinicianPortal\_Automation\_TP2186**” folder in root drive and open the “**readme.txt**” file.
11. Ensure that the contents of the “**ClinicianPortal\_Automation\_TP2186**” folder is as per the “**readme.txt**” file.

## Test Setup

The purpose of this section is to describe the setup of the files that are required for executing the test automation scripts:

A folder named "ClinicianPortal\_Automation\_TP2186" is created in the root drive(c:\) where this "ClinicianPortal\_Automation\_TP2186.zip" is extracted. Navigate to the “ClinicianPortal\_Automation\_TP2186” folder in the test PC and open the “readme.txt” file and verify the contents of the “ClinicianPortal\_Automation\_TP2186” folder, as per the readme.txt file.

Ensure that contents of the folder " \ClinicianPortal\_Automation\_TP2186" contain the following:

* 1. ‘Configuration’ folder
  2. ClinicianPortalAutomation.exe
  3. Nx2meConnected.Health v1.0.0.exe
  4. DllReg.bat

Ensure that contents of the folder "Configuration" contain the following:

a. Driven\_Data’ folder

b. ‘libs’ folder

c. ‘TestToolData’ folder

d. testng.xml

Ensure that the contents of this folder "\Driven\_Data" should contain the following:

a. Portal\_Drive\_Data\_1.xlsx

Ensure that the contents of this folder "\libs" should contain the ‘tools’ folder.

Ensure that the contents of this folder " tools " should contain the following:

1. ‘AutoItX4Java.jar’
2. ‘jacob.jar‘.
3. ‘jacob-1.19-x64.dll’
4. ‘jacob-1.19-x86.dll’

Ensure that the contents of this folder "\TestToolData" should contain the ‘TP2186’ folder:

1. TP2186’

Ensure that the contents of this folder " \TP2186’ " should contain the following:

1. ‘create\_upload\_files’ folder
2. ‘Test\_Data’ folder

Ensure that the contents of this folder " \create\_upload\_files " should contain the following:

1. CreateFlowsheet\_to\_generate\_alert\_0.xml
2. CreateFlowsheet\_to\_generate\_alert\_1.xml
3. CreateFlowsheet\_to\_generate\_alert\_3.xml
4. CreateFlowsheetFromFilesTask\_for\_5\_fs\_with\_attachments.xml
5. CreateFlowsheetFromFilesTask\_for\_10\_fs\_Unconfirmed.xml
6. CreateFlowsheetFromFilesTask\_for\_40\_fs\_Confirmed.xml
7. CreateFlowsheetFromFilesTask\_for\_50\_fs\_Confirmed.xml
8. DashboardPatientSettingsTask1.xml
9. DashboardPatientSettingsTask2.xml
10. DashboardPatientSettingsTask3.xml
11. DbDeleteTask.xml
12. PdmpPatientSettingsTask1.xml
13. PdmpPatientSettingsTask2.xml
14. PdmpPatientSettingsTask3.xml
15. PdmpTwoWayMessagingTask.xml
16. Upload\_Alert0.cfg
17. Upload\_Alert1.cfg
18. Upload\_Alert3.cfg
19. Upload\_XML\_10\_Unconfirmed\_fs\_Files.cfg
20. Upload\_XML\_40\_Confirmed\_fs\_Files.cfg
21. Upload\_XML\_50\_Confirmed\_fs\_Files.cfg
22. Upload\_XML\_with\_attachments.cfg
23. UploadAttachmentTask.xml
24. UploadFlowsheetTask.xml

Ensure that the contents of this folder " \Test\_Data " should contain the following:

1. Chrysanthemum.jpg
2. MyAttachments.txt
3. PostTxAssessmentSection\_alert0.txt
4. PostTxAssessmentSection\_alert1.txt
5. PostTxAssessmentSection\_alert3.txt
6. PreTxAssessmentSection\_alert0.txt
7. PreTxAssessmentSection\_alert1.txt
8. PreTxAssessmentSection\_alert3.txt
9. PreTxCommentsFile.txt
10. RxSettingsSection.txt
11. RxSettingsSection\_alert0.txt
12. RxSettingsSection\_alert1.txt
13. RxSettingsSection\_alert3.txt
14. TxDataSection.txt
15. TxDataSection\_alert0.txt
16. TxDataSection\_alert1.txt
17. TxDataSection\_alert3.txt

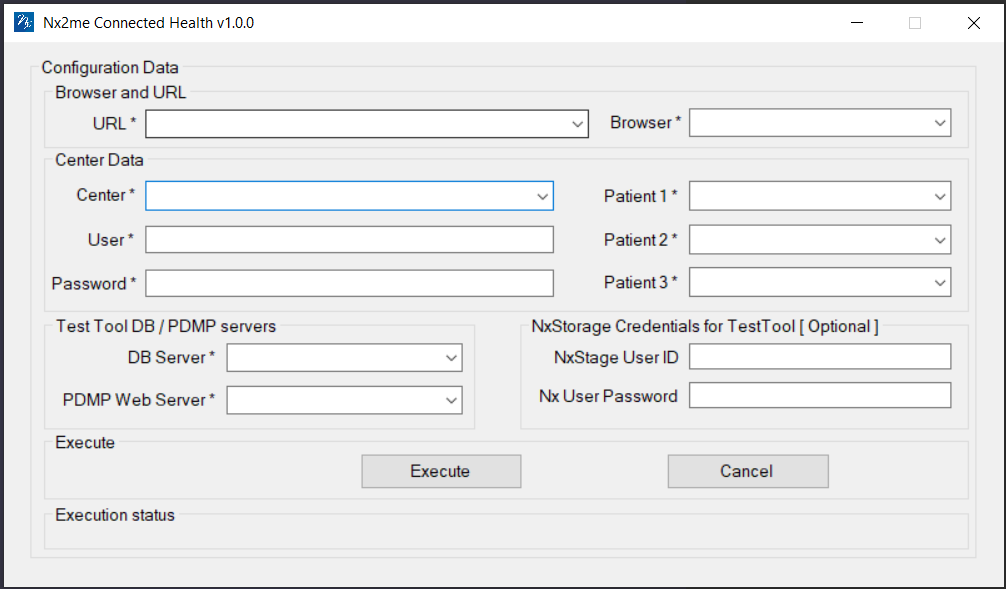
The Nx2me Clinician Portal – Test Automation of TP2186 performs automation in three stages.

* + User interface verification
  + Functionality verification
  + Flowsheet Statistics verification

# Nx2me Clinician Portal UI

The purpose of this section is to describe the functionality of the user interface application that is developed for launching the selenium test scripts.

A windows form is developed with the following user controls:



Using the above UI, the user can enter the input data and click ‘Execute’ button to start the automated testing of Nx2me Clinician Portal.

The excel sheet of ‘Portal\_Drive\_Data\_1.xlsx’ contains the following tabs:

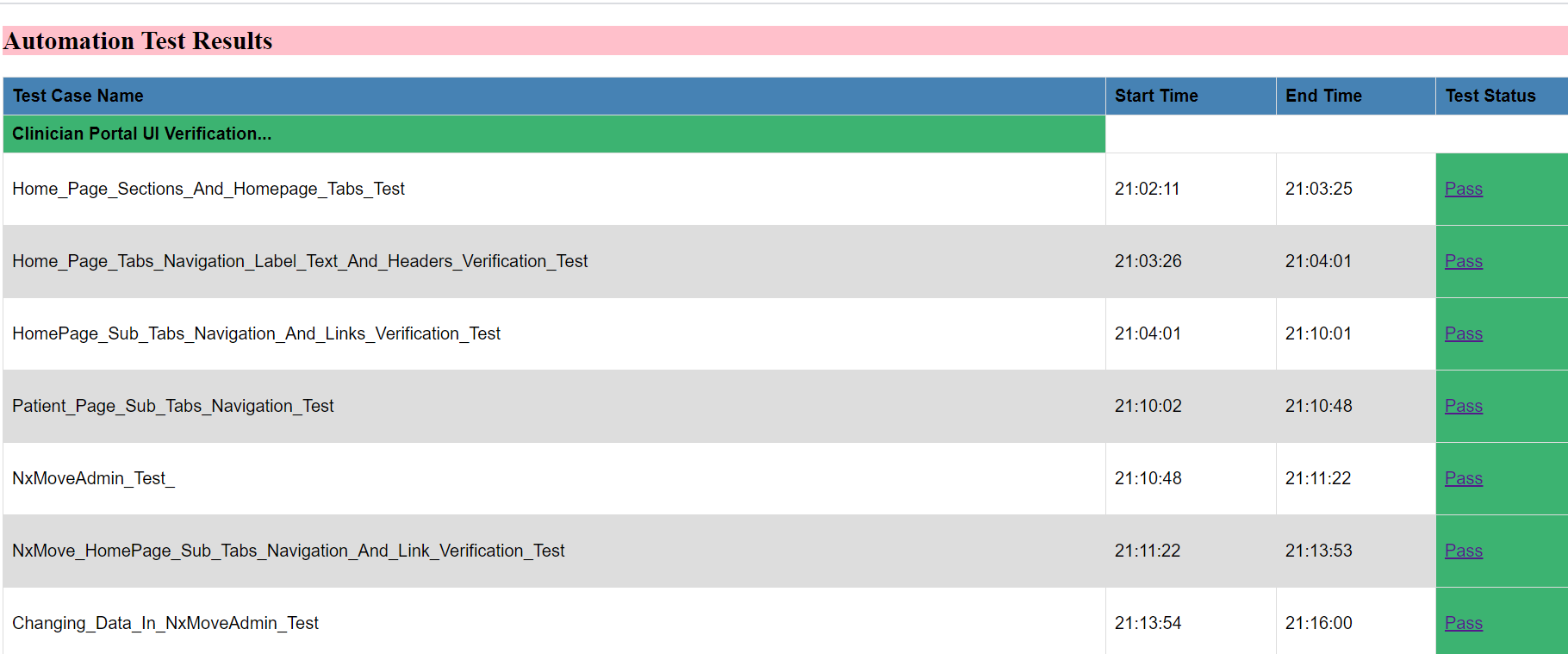
* + UI\_Functional\_Verification\_Data
  + Patient\_TX\_Settings\_
  + Patient\_Alerts\_Criteria

The ‘ UI\_Functional\_Verification\_Data’ tab contains all the inputs that are used for performing User interface and functionality verification of the ‘Nx2me Clinician Portal’ web application. Similarly, the ‘Patient\_TX\_Settings’ and Patient\_Alert\_Criteria’ tabs contains the three sets of input data that are used for performing Patient Treatment Settings and Alerts change settings.

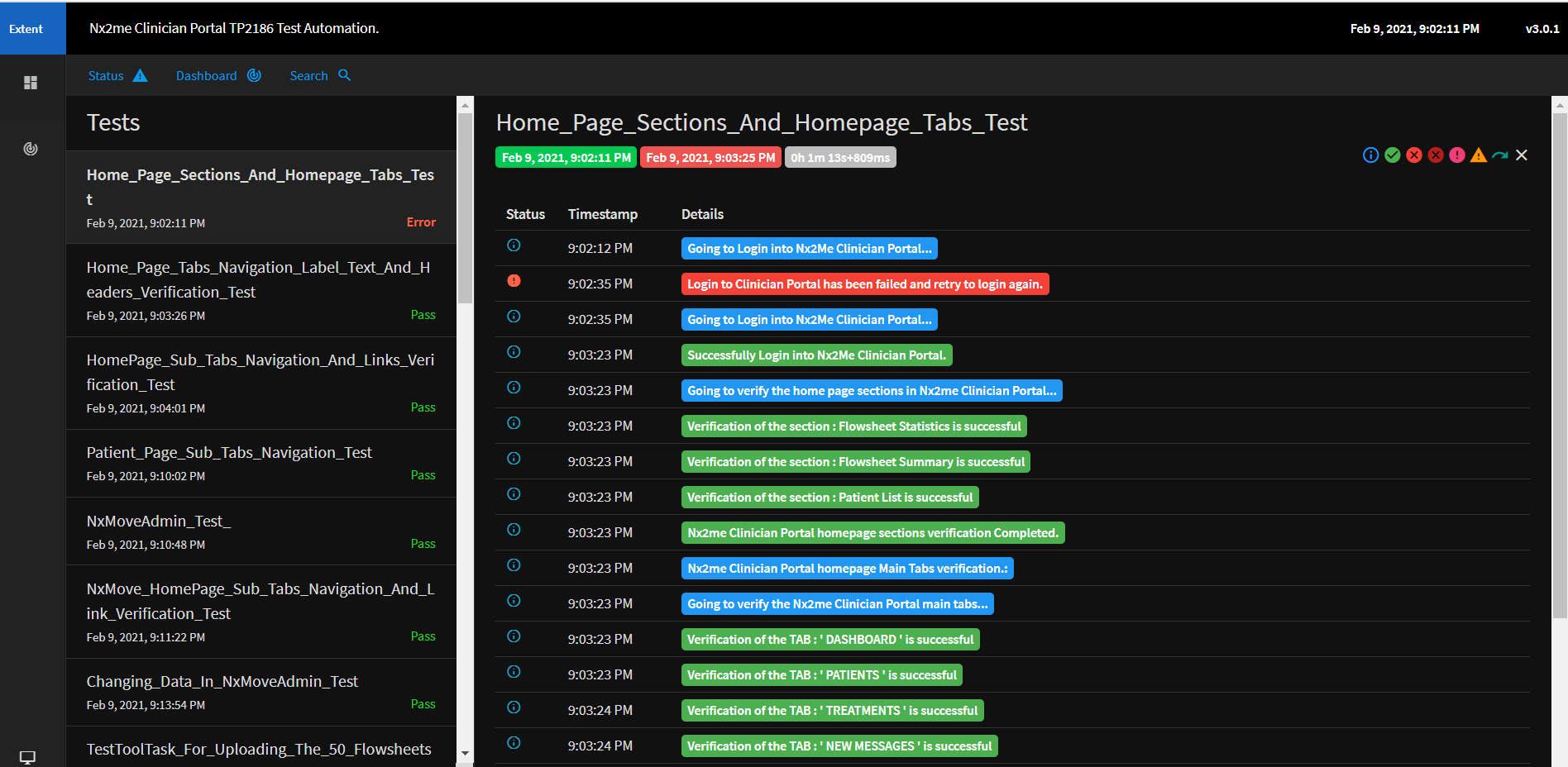
The Selenium test scripts retrieves the values from the above excel sheet while performing the user interface and functionality verification of all the web pages.

# User Interface Verification Test Case

The following is the summary list and the details of the test cases present in Nx2me Clinician Portal Automation – Running Status report:



And the above 7 test cases are the user interface test cases for all the web pages in the Nx2me Clinician Portal. The user needs to click the Hyper link of any test case displayed as ‘Pass’ in the above ‘running status report’ to access the following details:



* 1. The information text that are highlighted in blue color contains the details of every test step from TP, that is going to be executed.
  2. The information text that are highlighted in green color contains the details that every test step from TP, has successfully completed its execution.
  3. The information text that are highlighted in purple color contains the details that the navigation from a main window to a child window has been successfully done.
  4. The information text that are highlighted in red color contains the details of the test step from TP, that has failed.
  5. The last information text that is highlighted in green color contains the status of the functional test case that it has successfully passed.
  6. The last information text that is highlighted in red color contains the status of the functional test case that it has failed.
* The test case ‘Home\_Page\_Sections\_And\_Homepage\_Tabs\_Test’ This test method will perform below actions:

1. Home page sections.

2. Home page tabs verification.

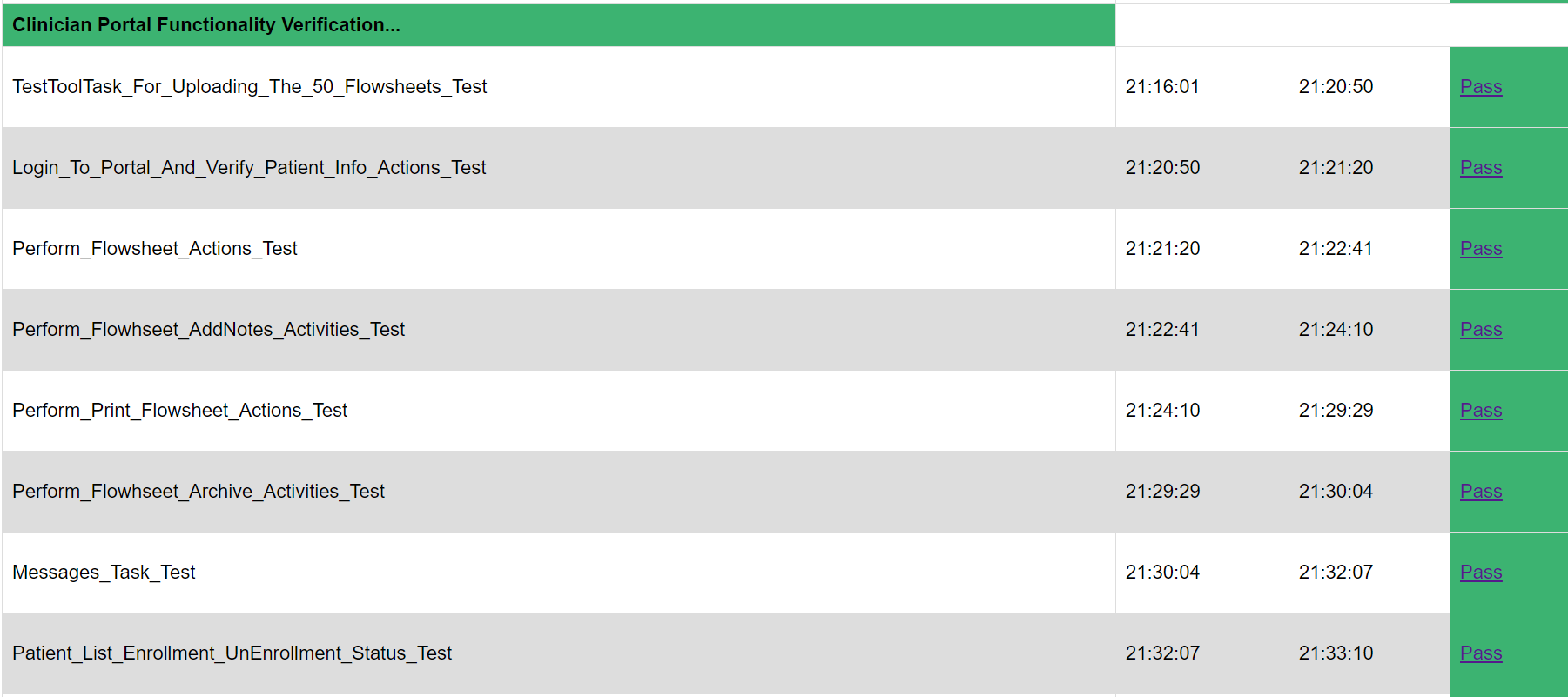
* The test case ‘Home\_Page\_Tabs\_Navigation\_Label\_Text\_And\_Headers\_Verification\_Test” This test method will perform Naviagation of all Dashboard Tabs, header and footer verification for all Tabs.
* The test case “HomePage\_Sub\_Tabs\_Navigation\_And\_Links\_Verification\_Test” This test method will perform Navigation of all the Dashboard Tabs and perform the child links test.
* The test case “Patient\_Page\_Sub\_Tabs\_Navigation\_Test” This test method will perform patients tab subtab navigation and verify the tabs texts.
* The test case “NxMoveAdmin\_Test\_” This test method will perform below actions:

1. NxMoveAdmin Header and footer text.
2. NxMoveAdmin child link tests.

* The test case “NxMove\_HomePage\_Sub\_Tabs\_Navigation\_And\_Link\_Verification\_Test” This test method will perform the navigation of all the main tabs and child link tests.
* The test case “Changing\_Data\_In\_NxMoveAdmin\_Test” This test method will perform the settings change in nxmoveadmin center, such as center access, messaging alerts etc,.

# Functionality Verification Test Case

The below test cases are the Functionality verification test cases for all the web pages in the Nx2me Clinician Portal.





* The test case “TestToolTask\_For\_Uploading\_The\_50\_Flowsheets\_Test” This test method will perform the upload of 50 flowsheets to portal using TestTool.
* The test case “Login\_To\_Portal\_And\_Verify\_Patient\_Info\_Actions\_Test” This test method will perform the below actions:

1. Patient information.

2. Flowsheet summary sections.

3. Default date range.

* The test case “Perform\_Flowsheet\_Actions\_Test” This test method will perform the below actions:

1. Flowsheet control verification.

2. Flowsheet status actions.

* The test case “Perform\_Flowhseet\_AddNotes\_Activities\_Test” This test method will perform the below actions:

1. Flowsheet Add notes.

2. Deleting center notes.

3. Center and Patient notes ICON test.

* The test case “Perform\_Print\_Flowsheet\_Actions\_Test” This test method will perform the below actions:

1. Flowsheet Print with / without attachments action and verification.

2. Alerts Verification.

* The test case “Perform\_Flowhseet\_Archive\_Activities\_Test” This test method will perform the below actions:

1. Flowsheet archive functionality.

2. Flowsheet pagination to different pages.

* The test case “Messages\_Task\_Test” This test method will perform the below actions:

1. PatientTwoWayMessaging task.

2. Create new message, and giving reply to same message.

* The test case “Patient\_List\_Enrollment\_UnEnrollment\_Status\_Test” This test method will perform the below actions:

1. Enrollment and Unenrollment status task.

2. Ascending or descending test.

* The test case “TestTool\_Task\_For\_Uploading\_Unconfirmed\_Flowsheets\_Test” This test method will perform the upload of Unconfirmed flowsheets to the portal using TestTool.
* The test case “Patient\_Page\_Sub\_Tabs\_Functionality\_Test” This test method will perform the patients tab subtab functionality, such as reports, messages, navigating to settings , summary and alerts.
* The test case “Patient\_Alerts\_Verification\_Test” This test method will perform the patients Alerts and Alerts criteria verify test.
* The test case “Alerts\_Change\_Test1” This test method will perform the patients Alerts criteria change test.
* The test case “Patient\_Settings\_Test1” This test method will perform the patients prescription settings change test.

# Flowsheet Statistics Verification Test Case

The below test cases are the Flowsheet statistics verification test for Nx2me Clinician Portal.

# 

* The test case “Flowsheet\_Statistics\_Test” This test method will perform the flowsheets panel statistics sections.
* The test case “Upload\_50\_Flwosheets\_And\_Do\_Flowsheet\_Statistics\_Test1” This test method will perform the uploading of 50 flowsheets to the portal app using the test tool and do the flowsheet statistics test (Changing flowsheet status, archive functionality).
* The test case “Upload\_10\_Unconfirmed\_And\_5\_Confirmed\_Flowhseets\_And\_Do\_Verification\_Test” This test method will perform the below actions:

Test Tool activities:

1. Upload 10 Unconfirmed Flowsheets to portal.

2. Upload 10 confirmed Flowsheets to portal.

3. Upload 5 new Flowsheets to portal.

Verification:

1. Unconfirmed, confirmed flowsheets status.

2. Flowsheet statistics panel verifictaion.

* The test case “Alerts\_Test\_For\_atert0\_1\_3\_Test” This test method will perform the below actions:

Test Tool activities:

1. Alert 0 generation.

2. Alert 1 generation.

3. Alert 3 generation.

4. Upload an unconfirmed flowsheet to PN2.

5. Upload an unconfirmed flowsheet to PN1

Verification:

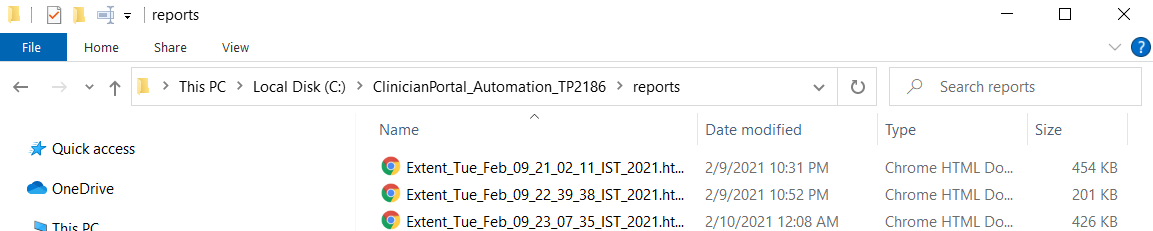
1. flowsheet Statistics Verification

2. Alerts settings and change settings.

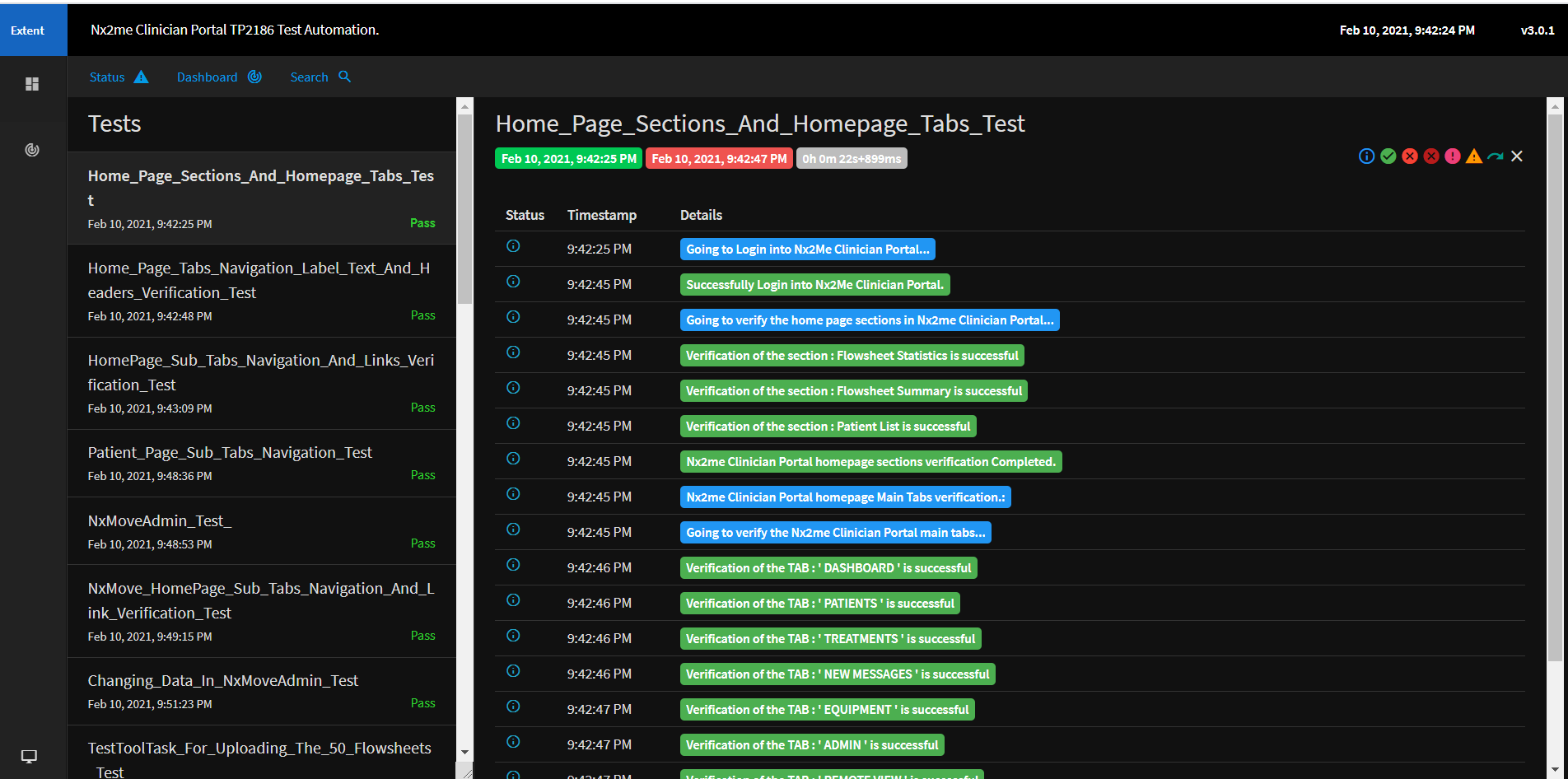
* The test case “DB\_Delete\_And\_PatientProvision\_Task\_Test” This test method will perform the DB Delete task on center, and provisioning the center and as well as provision the patient.

# Result Verification

The following is the summary of results that will be displayed in the extent HTML report, after completing the execution of the functionality verification test cases and UI verification test case and given below are the steps to access the report:



Navigate to the ‘\ClinicianPortal\_Automation\_TP2186\reports\’ folder and open the extent report HTML file and click on the ‘Pass’ Hyper link of the test case to access the below details:



# Source Code

The Nx2meConnected.Health v1.0.0 Automation Application is developed using C# in Visual Studio 2019 environment. The UI is a windows form that uses Microsoft Excel Interop Assemblies to retrieve and save entries in Microsoft Excel document in Dosing Calculator project.

The Nx2me Clinician Portal Selenium Test Scripts is developed using Java and Selenium with TestNG framework in Eclipse IDE – 2020.

The following path contains the source code of the Selenium test scripts in SVN:

http://subversionval.nxstage.com:8088/svn/NxAutomation/trunk/Clinician Portal/TP2186

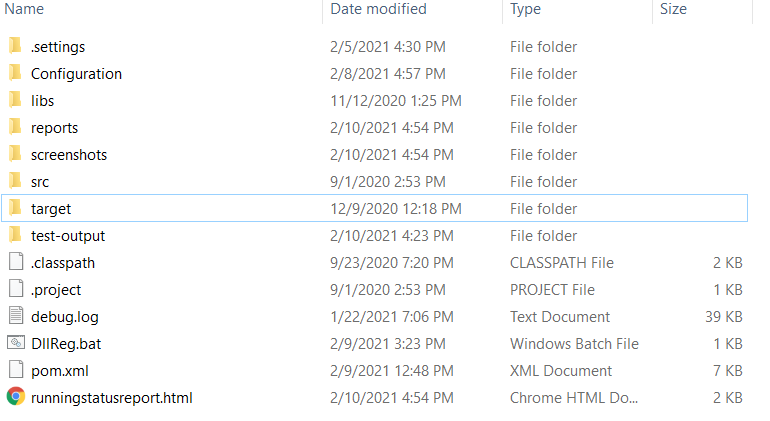
And the below path contains the source code of the reusable methods defined in AppUtil.jar in SVN:

http://subversionval.nxstage.com:8088/svn/NxAutomation/trunk/AppUtil\_Supportive\_JAR/AppUtil

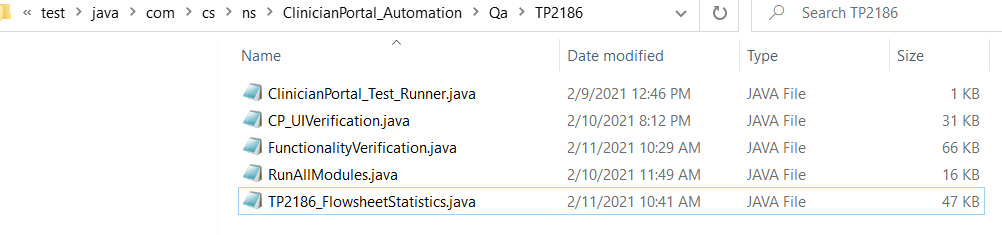
## Source Code Structure

### Java and Selenium

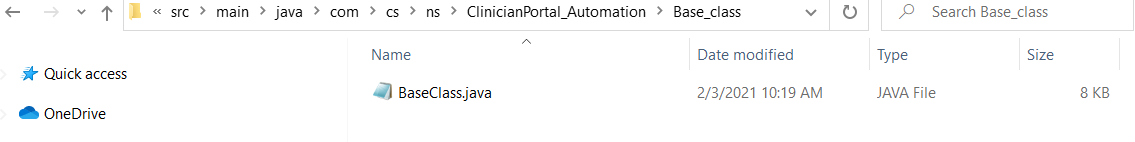
1. Create a new directory. This directory will be referred to in the remainder of these instructions as the “Working Directory”. And this working directory contains the below folders:



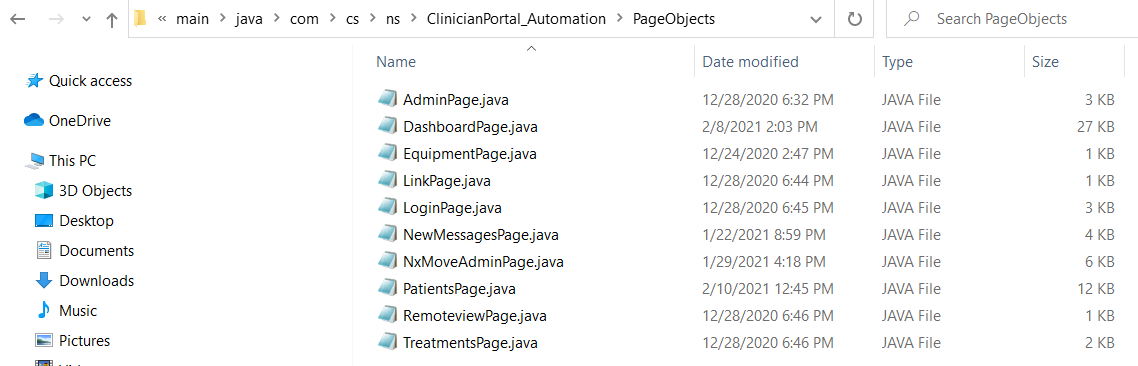
1. The file ‘pom.xml’ contains the list of all the dependencies using the Nx2me Clinician Portal project.
2. The ‘src’ folder contains the selenium scripts of the user interface, functionality and flowsheet statistics verification test classes.



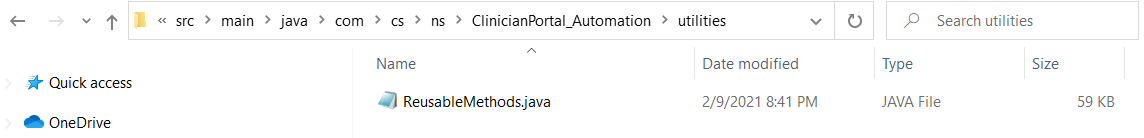
1. The ‘Base\_class folder contains ‘BaseClass.java’ where the BaseClass is inherited from the ‘Listener’ class and the static variables are stored:



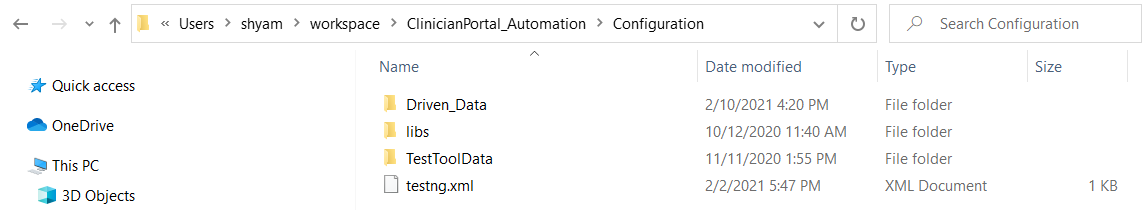
1. The ‘PageObjects’ folder contains all the web elements stored java classes.



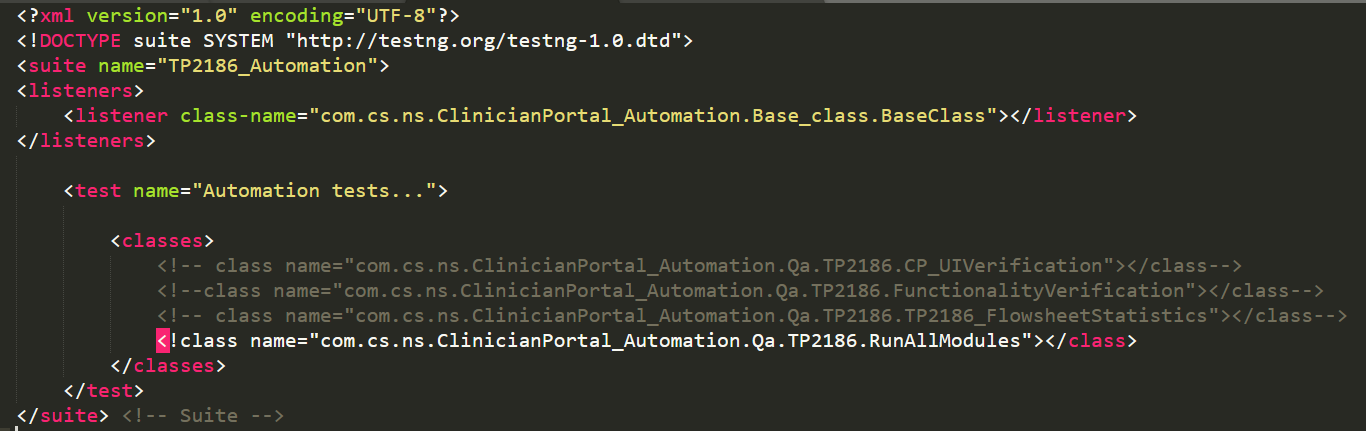
1. The ‘utilities’ folder contains the ReusableMethods.java class..



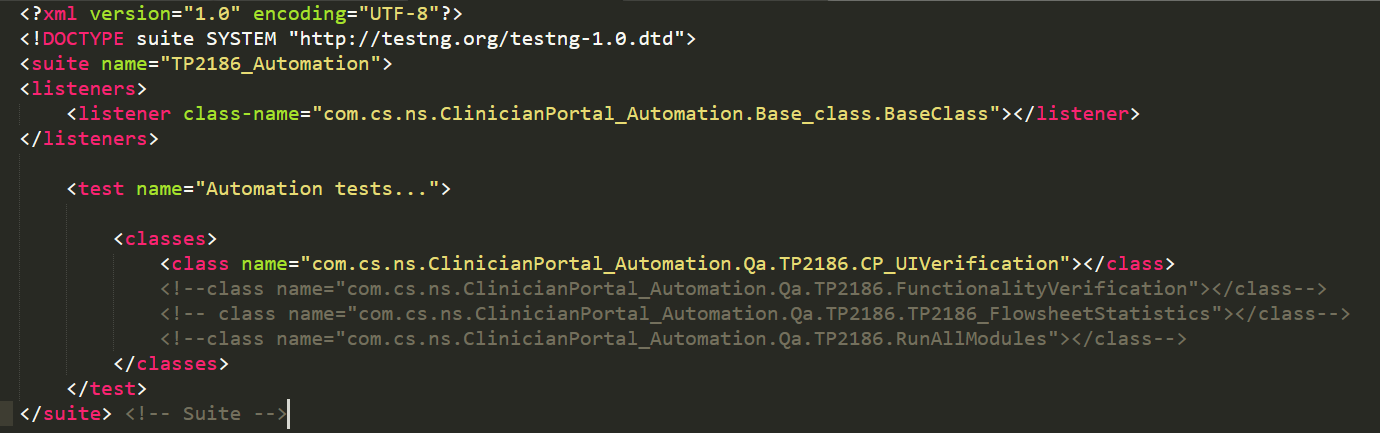
1. The ‘configuration’ folder contains the below folders:

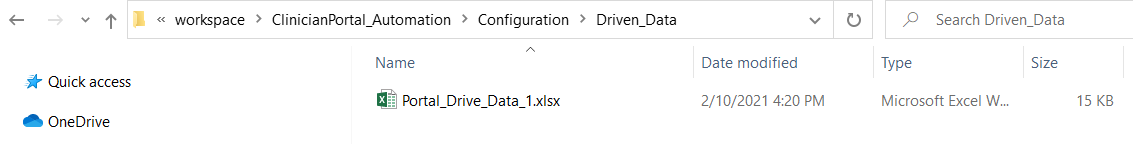


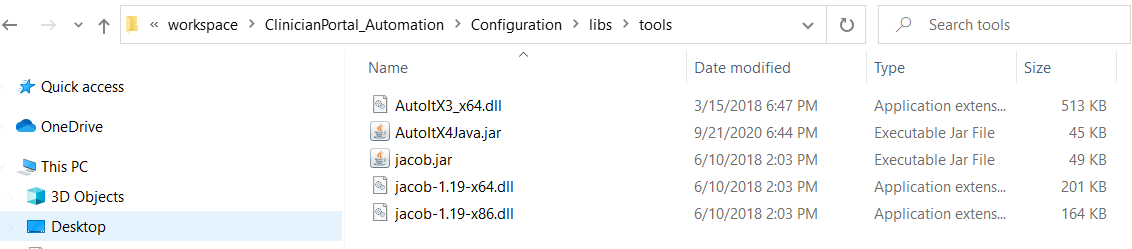
The Nx2me Clinician Portal Selenium scripts TP2186 is having three (3) sections (“ CP\_UIVerification”, “FunctionalityVerification” and “FlowsheetStatistics”,) in order to run all the three modules, enable the tag of **RunAllModules** in testing.xml as shown below.

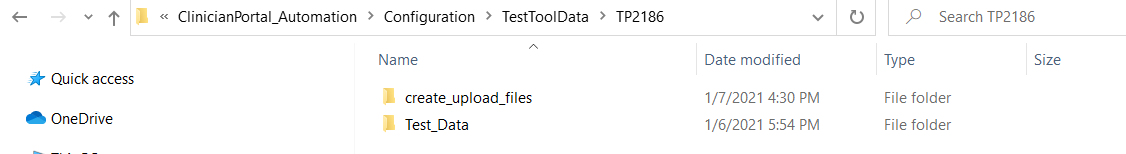


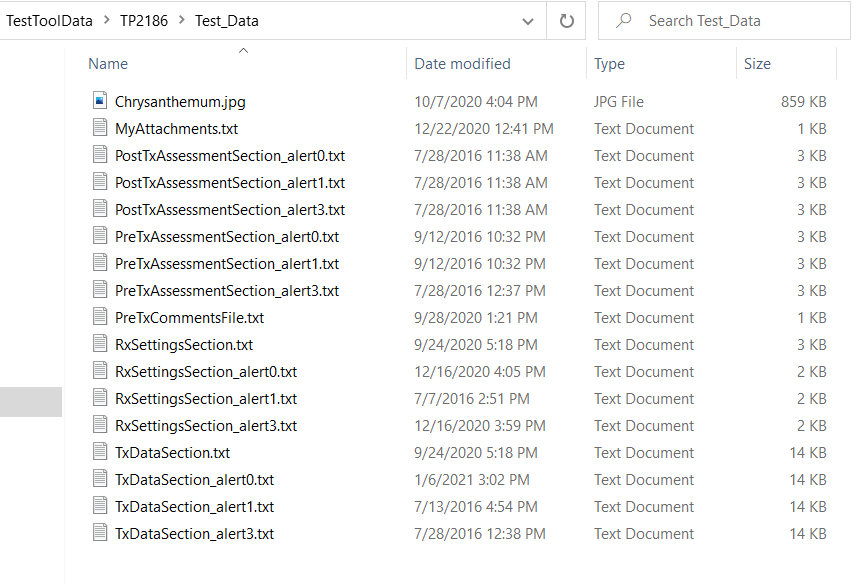
For individual module execution like “ CP\_UIVerification”, “FunctionalityVerification” and “FlowsheetStatistics” enable the single module and save the testing.xml as shown below.

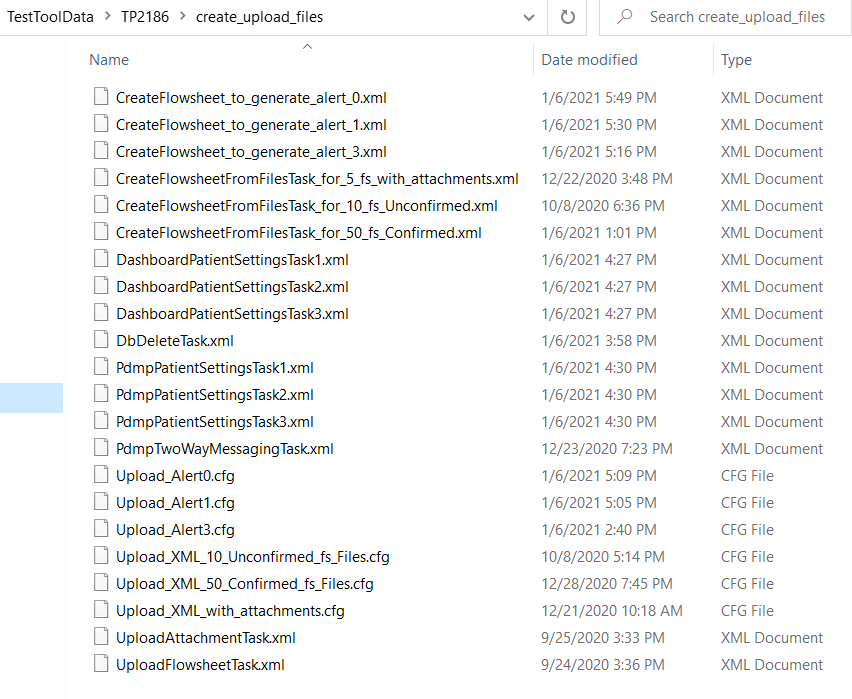










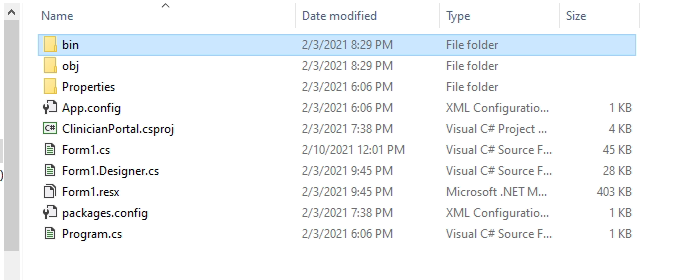


1. The ‘’libs’ folder contains the ‘AppUtil’ jar where the reusable methods for developing the selenium test scripts are stored:

| Table – AppUtil Module Methods | | |
| --- | --- | --- |
| **File Name** | **Reusable Methods** | **Selenium Test Scripts** |
| Input.java | setExcelPath(), setBrowser(), urlNavigate(), getDriver(), clickOn(), textBoxEntry(), clearTextBoxLabel(), isElementPresent(), getLabelText(), getAttributeValue(), switchChild\_To\_MainWindow(),dropDown(), mouseOver(), timestamp(), getDateTime(), getTime(), datePicker\_Entry(),  waitUntilElementClickable(),getCSSValue(),  scrollDown(),scrollUp(), isClickable(),  substract\_DaysFromTodaysDate(),  dateDifference(),  driverInstanceKillProcess() | All the test cases. |
| Verify.java | checkIfLabelEnabled(), checkIfLabelDisplayed(),  checkIfLabelPresent(),  checkIfChkBoxEnabled(),  checkIfChkBoxRdoBtnSelected () |
| Excel.java | excelData(), getDataFfromExcel(), writeToExcel() |
| ExtentreportsGenerator.java | test.log() |
| StatusHTML.java | htmlTagUpdate(),setStatusOfHTML(),  htmlTitleTag() |  |
| TestTool.java | testtoolDataConfiguration(),  testtoolTaskExec(),  setTestToolDBServer(),  setTestToolPDMPWEBServer(), |
| XMLHandler.java | updateXmlTagValue() |

### User Interface Application

1. The ‘Nx2me Clinician Portal’ folder contains the ‘Windows Form’ of the UI and the source code file is: ‘Form1.cs’:



## Updating Code Guidelines

The below documents are referred:

SOP-04C-013Software Coding Standards and Guidelines.doc

## Reference

The below documents are referred:

|  |  |
| --- | --- |
| **No.** | **Document Name** |
| [1] | SSS0144 Nx2me Clinician Portal Software Requirements Specifications |
| [2] | TP2186 Nx2Me Clinician Portal Browser Compatibility Verification Test Procedure |
| [3] | TP2384 Nx2Me Clinician Portal Browser Compatibility Automated Verification Test Procedure |

# Environmental Issues and Solutions

The following environment is used for development and testing:

|  |  |  |
| --- | --- | --- |
| **No.** | **Item Description** | **Version / Speed / Permissions** |
| 1 | Windows | 10 |
| 2 | Java | 14.0.1 or above |
| 3 | Selenium | 3.14.0 |
| 4 | Eclipse IDE for Java Developers | 2020-09 (4.17.0) |
| 5 | Microsoft Visual Studio Enterprise 2019 (C#) | 16.7.7 |
| 6 | Microsoft .NET Framework | 4.8.03752 |
| 7 | Internet Speed | 2 Mbps |
| 8 | Root Drive (c:\) | Read and Write access |

# Script Update Policy

|  |  |  |
| --- | --- | --- |
| **No** | **Item Description** | **Update** |
| 1 | Selenium Scripts | Functional change done in any web page, UI modifications done in any web page in Nx2me Clinician Portal Application. |
| 2 | Nx2me Clinician Portal Automation Application | Any new feature required to be built in the Windows Form of Automation application |

# Revision History

|  |  |  |
| --- | --- | --- |
| **Revision** | **ECO No.** | **Description of Change** |
| A | DCO23879 | Initial Release |