Description: Start site

Player Club Booth Portal Automation Guide

Draft Version

Contents

[ **Document Control** 3](#_Toc173490247)

[ **Objective** 4](#_Toc173490248)

[ **Installation and Prerequisites.** 4](#_Toc173490249)

[ **Environment Details** 4](#_Toc173490250)

[ **Java Classes Structure** 4](#_Toc173490251)

[ **Test Data and Test Cases:** 6](#_Toc173490252)

[ **Configuration:** 6](#_Toc173490253)

[ **TestNG XML and Script Execution** 7](#_Toc173490254)

[ **Parallel Execution support** 8](#_Toc173490255)

[ **Reports** 9](#_Toc173490256)

[ **POM** 11](#_Toc173490257)

* **Document Control**

|  |  |  |  |
| --- | --- | --- | --- |
| **Version** | **Date** | **Author** | **Comments** |
| DRAFT V1 | 15/01/2024 | Yogesh Tayal | Initial draft |
| DRAFT V2 | 7/02/2024 | Yogesh Tayal | Document updated with Extent report, screenshots description based on the feedback received on 31 Jan |
| DRAFT V3 | 2/08/2024 | Yogesh Tayal | Added more information for manual execution, also added Parallel execution support. |

* **Objective**

The objective of this document is to explain the automation framework setup and guide the user to execute the scripts and analyzing the results.

* **Installation and Prerequisites.**

The following software need to install.

1. JDK 1.8+ (Ensure that the Java class path is properly set)
2. Eclipse for Java developers (latest version)
3. TestNG plugin for eclipse
4. Maven plugin for eclipse (Default it should come with eclipse, if not please install)

## **Environment Details**

**URL:** <https://10.222.80.152:886/portal/#/applications> (This URL is configurable and present in config.properties)

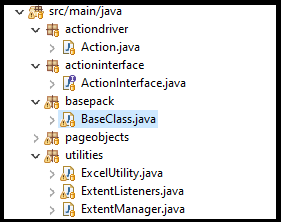
**Browser:** Chrome

**Login Details:** Site – SiteOne, User ID – qaautomation, Password- Test@1234

**Script Path:** [Advantage-Web-Access.PlayersClub/Testing/UI Testing/PlayerClubBooth at feature/Automation-UI-Testing · igt-casino/Advantage-Web-Access.PlayersClub (github.com)](https://github.com/igt-casino/Advantage-Web-Access.PlayersClub/tree/feature/Automation-UI-Testing/Testing/UI%20Testing/PlayerClubBooth)

## **Java Classes Structure**

1. All the framework common actions method to be performed in project are placed under the maven’s **src/main/java** folder.



a) The **actiondriver** Package has abstract Action.java class is designed to encapsulate common actions or user interactions with web elements. The goal is to create a reusable set of methods that represent various actions a user might perform on a web page.

It Allows you to perform user interactions like keyboard action, mouse actions and many more.

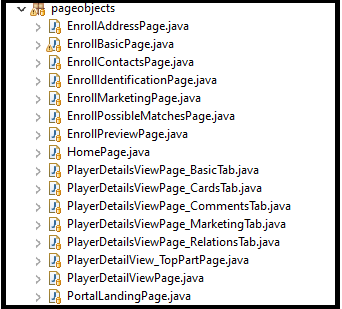
b) The **actionInterface** package has ActionInterface.java class which defines a set of common methods that involve interacting with web elements. Classes implementing this interface are expected to provide concrete implementations for each of these methods.

c) The **basepack** package has Baseclass.java is a central class that contains common functionalities used to initialize the WebDriver, settings, and load configurations shared across multiple test classes. Test classes can then extend this base class to inherit the common settings.

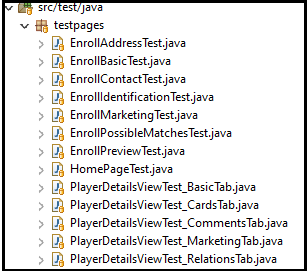
d) The **excelUtility** (placed in utilities folder) contains methods to interact with Excel files. This utility class uses Apache POI library, Java library for reading and writing Microsoft Office files, to handle Excel interactions.

e) The **ExtentManager** and **extent Listener** files are used for custom report generation.

1. Since we are following POM design pattern so, all the page classes are under “**pageobjects**” package, and these are under maven’s src/test/java folder. These classes are having the locators and functionalities of the respective pages.



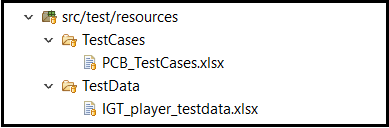
1. All test class pages files are under the package “**testpages**” which is under maven’s src/test/java folder.



## **Test Data and Test Cases:**

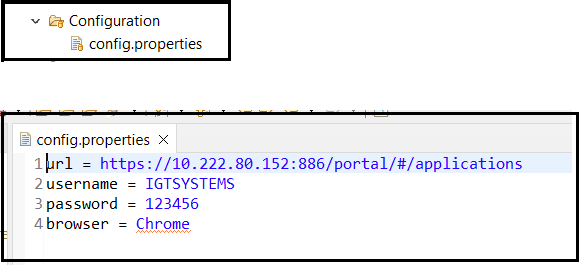
All the test resources like Test cases and Test Data are placed under “src/test/resources” folder.

* PCB\_TestCases.xlsx sheet contains all the testcases covered for respective user story.
* IGT\_Player\_testdata.xlsx contains test data to trigger automation suite.



## **Configuration:**

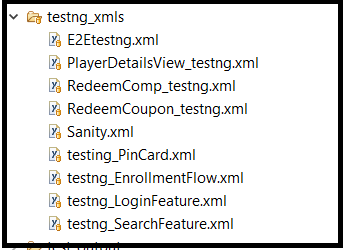
Configuration related to URL, username and password, browser settings can be configured from **config.properties** file under Configuration folder.



## **TestNG XML and Script Execution**

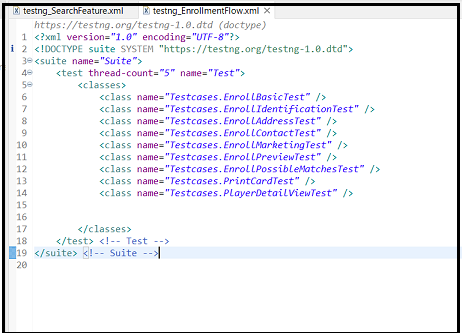
1. The TestNG xml file are created to execute the complete suite based on the features and placed in folder **testng\_xmls**.

* For **End to end complete testcases** can be executed **from E2Etestng.xml**
* For **Sanity test**, it can be executed from **Sanity.xml**
* **Search a player** feature can be execute from **testng\_SearchFeature**.xml
* **Enrollment of player** feature can be executed from **testng\_EnrollmentFlow.xml**
* **Login** to portal feature can be execute from **testng\_LoginFeature**.xml

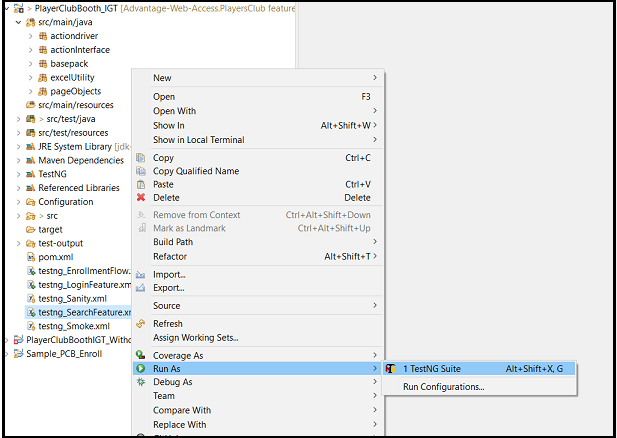


1. To run the test scripts in local open the testng.xml let say **testng\_SearchFeature.xml**

and right click on the **testng\_SearchFeature.xml** ->Run As -> TestNG Suite



To execute Script:



## **Parallel Execution support**

* For executing Testcases parallelly , update the testing.xml thread count from 1 to 3.



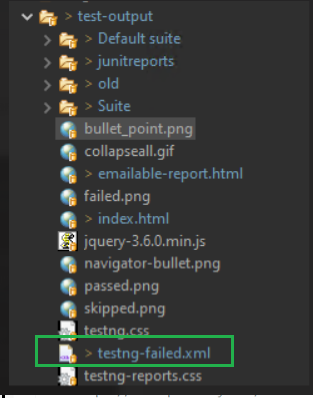
## **Reports**

1. Once the execution is over user can see the **Extentreport.html** under test-output folder.

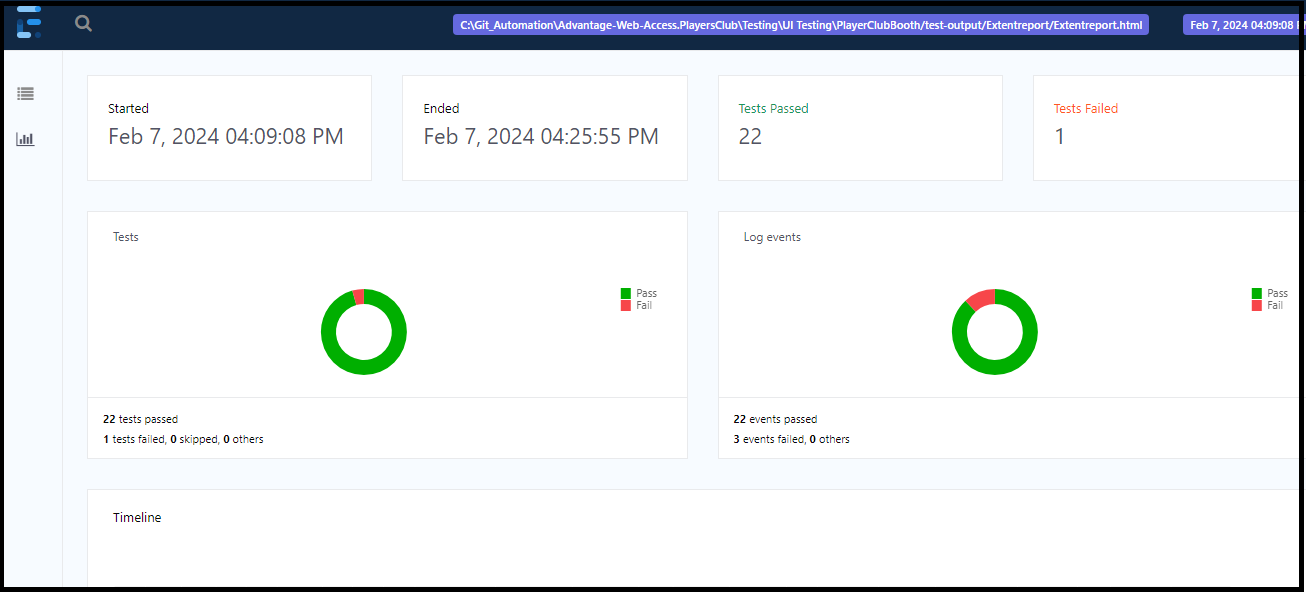
A screenshot of a computer

Description automatically generated

Note - In case few testcase got failed for some reason, we can re run the failed testcases by right clicking and Run As -> TestNG Suite



1. To open this report, right click on extentreport.html and open with Web Browser. It will show all the test script execution status.



A screenshot of a computer

Description automatically generated

## **POM**

Maven’s pom.xml file used to have all the external jar dependencies.

