|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Test Scenario ID** | | dfwnet | | | | **Test Case ID** | | dfwnet-tc-007 | | |
| **Test Case Description** | | Each component downloaded from GitHub compiles and works on Linux | | | | **Test Priority** | | High | | |
| **Pre-Requisite** | | Linux OS. Dotnet core framework installed | | | | **Post-Requisite** | |  | | |
| **Technology Stack** | | .NET Core 2.1 | | | | **Checked by** | | José Manuel Sánchez | | |
| **Test Procedural Steps** | | | | | | | | | | |
| **Step** | **Action** | | **Inputs** | **Expected Output** | **Actual Output** | | **External Links** | | **Remarks** | **Test Result** |
| 1 | Download devon4net development branch | | Clone devon4net repository | A oasp4net folder with framework distribution inside | A folder with name the “oasp4net”. The folder has the devon4net distribution | | https://github.com/oasp/oasp4net.git | |  | **OK** |
| 2 | Go to Modules folder and run on a command prompt build.cmd | | build.cmd command | A new folder “results” is created with a text file for each module compilation | A new folder “results” is created with a text file for each module compilation | |  | |  | **OK** |
| 3 | Go to results folder and check every .txt which contains the compilation log results for each module. | | A text file for each module compilation | Each file has the output log for each module compilation. Each compilation must end with no error results | Each log file revised. No errors where found | |  | |  | **OK** |

This document is a template for tracking the test case performed for a specific scenario.