**Context Injection for Steps Reusability with SpecFlow and Selenium WebDriver.**

**<Draft Ver-1.0>**

Automated Acceptance testing is important milestone in Acceptance Test Driven Development. The automation tools like SpecFlow and Selenium playing an active role in automated acceptance driven test scenarios in the Agile projects.

**Problem Statement:**

In the Acceptance Test Driven Development with SpecFlow and WebDriver, throwing exception during the test run, which indicates the non-availability of WebDriver to perform the Test. Since, once the instance of WebDriver shared across the step definitions for execution, WebDriver availability is getting nullified and getting "WebDriver" not available exception.

**Solution:**

SpecFlow have the luxury of "Context Injection" feature, which support to share the WebDriver instance across the step definition.

**Context Injection:**

The Context Injection feature will instantiate and inject class instance for the scenarios. It groups the shared state to Context-classes, and inject them into every binding class that is interested in that shared state. The context injection designed to share data between two step definitions. The two most important rule for Context Injection are

* The life time of an injected object is limited to the scenarios execution. If the injected object is implemented IDisposible, they will be disposed after the scenario execution.
* Within one scenario execution, same instance of the object will be shared across the step definitions.

**Implementation of Context Injection with SpecFlow and Selenium WebDriver.**

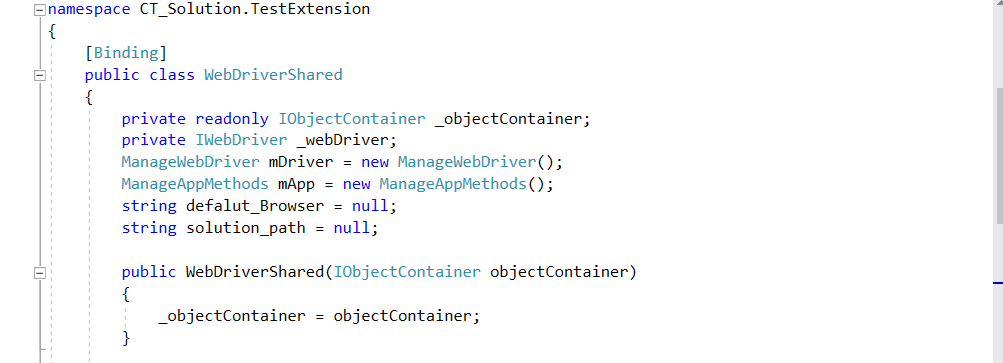
The below steps are mandatory for the implementation of Context Injection. The main step involved in the

Context Injection is

1. Create Class with IObjectContainer as a Constructor.
2. Create StepDefinition class with a Constructor, which will take WebDriver as a parameter
3. Implementation of Context Injection in a StepDefiniton.

**Step 01: Create a class which takes IObjectContainer as a constructor.**

1. Create a class with IObjectContainer and Register the same.
2. Add BeforeScenario to create the WebDriver Instance.
3. Add AfterScenario to Close and Dispose the Driver.

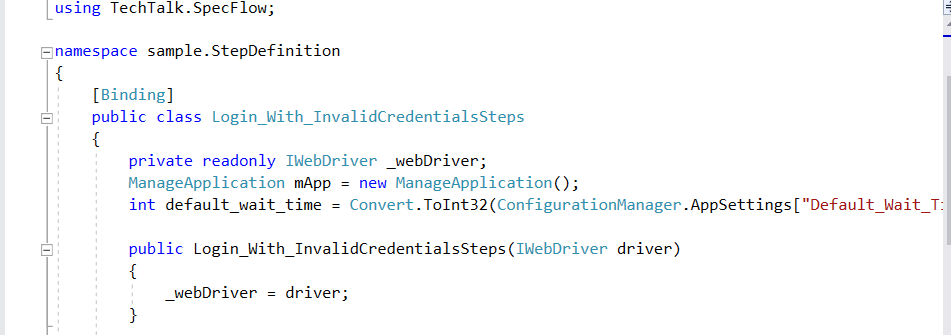




**Step 02: Create StepDefinition class with a Constructor which will take WebDriver as a parameter to enable access to WebDriver. Sample of Context Injection:**



**Refer Sample of a Full Step Definition.**



**Conclusion**

The context injection helps to avail the WebDriver across the StepDefintions to run the automated test scenarios.