**ATDD Automation Framework – “How to” Guide**

**How To setup your Automation Project**

-----------------------------------------------------------------------------------------------------------------------------------------

**Step** 1: Install Visual Studio 2017/2019

**Step 2**: Open Visual Studio -> Go to “Tools” -> Click on “Get Tools and Features…” ->Click on “Individual components” -> Select “GitHub extension for Visual Studio” Check box -> Click on “Modify” button

**Step 3**: Open Visual Studio and click on “Continue without Code”-> Sign In to Visual Studio ->Click on “View” Tab -> Select “Team Explorer” -> In Team Explorer click on “Manage Connections”  button -> select connect to github -> go to “Github Enterprise” ->Enter “github.optum.com” and provide your msid credentials , then click on clone and enter repository “<https://github.optum.com/BOS/QE-Automation>” and clone the repository.

**Step 4**: open the solution ATDD\_Automation.sln and go to Solution Explorer.

**Step 5**: For VS 2019 version, Go to Extensions -> Click on Manage Extensions -> Search For Specflow -> Download Specflow -> After download close all instances of Visual Studio to reflect the installation

For VS 2017 version, Go to Tools -> Click On Extensions and Updates -> Search For Specflow -> Download Specflow -> After download close all instances of Visual Studio to reflect the installation

**Step 6**: Open Visual Studio -> Go to Team Explorer -> Branches. Default branch is Masters.

Click on New Branch -> Create your own branch with the name in format Automation\_FirstName -> Click on Create Branch(Keep the checkout branch check box as selected)

When you create a branch it gets created in your local, you need to push it to server (GitHub).

For that Select your branch -> Right Click on your branch and click on “Push Branch”.

**Step 7**: Click on Build and select “Build solution” and wait for build to be successful.

**How To Check-in/Check out Code:**

-------------------------------------------------------------------------------------------------------------------------------

**Step 1**: Once done with your code changes, build your project in order to insure there are no errors.

**Step 2**: Once build is successful, check in your code as below process:

* Goto Team Explorer -> Home -> Changes. All your changes will be listed there. Verify the files which you have changed. If any other files that you have not changed are listed, check the changes in those files by selecting the option “Compare with Unmodified”. If changes are not valid, do not select those files for “Stage”.
* Select the required files for stage, and click on “Stage”.
* Write down the commit message and click on “Commit Staged” button.
* Click on “Sync” and click ‘push’ link to push your changes.
* After push is successful, open the GitHub repository URL -> your Branch -> create “New Pull Request” -> select a reviewer -> then click on “Create Pull Request”
* Next click on Merge Pull Request. It will merge your code changes to masters.

**Step 3**: In order to pull the current changes from server master branch to your branch, follow below process:

* Goto Team Explorer, right click and checkout master branch and right click and select pull option. Local master is updated with server master.
* Next checkout your branch and merge local Master branch changes to your local branch.

**How to create a new feature file in ATDD:**

------------------------------------------------------------------------------------------------------------------------------

**Step 1:**

To create a new feature file, right click on Feature File folder of SalesForce\_Automation project and select Add New Item. Select Specflow Option and Specflow Feature File.

**Step 2:**

When new feature file will be created then one default scenario will be present. Remove existing scenario and create new scenario following Gherkin format. Then right click on feature file and select “Generate Step Definitions” to create step definition for the scenario steps. Then click on Generate to create step definition file and select the path of file under StepDefinition Folder.

**Step 3:**

The StepDefinition file that created under StepDefinition Folder is having implementation of scenario steps. Remove the statement “ScenarioContext.Current.Pending();” which will be present default for each scenario step implementation and add automation code accordingly.

Notes: **Including a new Feature File**

When adding a new feature file to project follow the below link-

<https://specflow.org/documentation/Generate-Tests-from-MsBuild/>

For any clarifications, support, reach out to :

* **Samantaray, Swagatika** [**swagatika\_samantaray@optum.com**](mailto:swagatika_samantaray@optum.com)
* **Srivastava, Shubham** [**shubham\_srivastava@optum.com**](mailto:shubham_srivastava@optum.com)