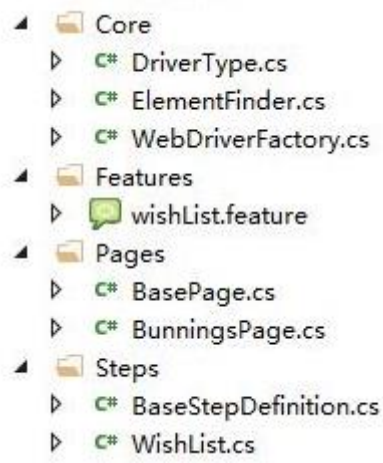
Report of technical test

This document is purposed to explain the **Technical Test** and what is implemented into the automation framework. This solution contains a test framework which is built from **Specflow, Nunit** and **Selenium WebDriver** and based on **page object model**.

✓ **The following coding structure:**



- **Code folder**: Initialize the web driver and method for finding elements
- **Features**: defined the test feature.
- **Pages**: defined the test pages.
- **Steps**: defined the test steps.

```
<appSettings>
    <add key="URL" value="https://www.bunnings.com.au" />
    <add key="DriverType" value="Chrome"/>
    <add key="searchKey" value="Paint"/>
    <add key="Added" value="Added"/>
    <add key="myWishList" value="Total"/>
    <add key="listUrl" value="https://www.bunnings.com.au/wish-lists/" />
  </appSettings>
</configuration>
```

The test data is added into App.config file as above.

✓ **The test execution as below:**

✓ **Achieved test scenarios:**

- Visit Search Result page with the search term “Paint” from Bunnings pages
- Randomly select one product
- Click the “Add to wish list” button
- Verify on the wishlist page.

Please refer to BunningsPage.cs file for methods and test steps in wishList.cs file.