AUT - <http://automationpractice.com/>

There are 5 scenarios with steps and expected result in the end. It doesn’t mean that you will have 5 auto-tests, you can decide by yourself how many auto-tests you have to implement to cover all scenarios.

**AP-1 Verify the ability to create an account**

1. Go to login page <http://automationpractice.com/index.php?controller=authentication&back=my-account>
2. Fill Email address input
3. Click Create an account button
4. Fill all required fields and click Register button

Expected result: Account was created

**AP-2 Verify the ability to login in account**

1. Go to login page <http://automationpractice.com/index.php?controller=authentication&back=my-account>
2. Fill Email address and Password inputs
3. Click Sign in button

Expected result: You was able to login

**AP-3 Verify the ability to add to auto-created Wishlist**

1. Login
2. Make sure that there is no Wishlist in account settings
3. Go to any product detail page and click Add to Wishlist button

Expected result: Wishlist was created automatically and your product is in the list

**AP-4 Verify the ability to add to your Wishlist**

1. Login
2. Create Wishlist in account settings
3. Go to any product detail page and click Add to Wishlist button

Expected result: Product was added to your Wishlist

**AP-5 Verify the ability to add to cart**

1. Login
2. Add 3 different products to cart
3. Go to cart

Expected result: All 3 products are in the cart and total is correct

**Technology stack**

* Programming language – Java
* Build and project management tool – Maven or Gradle
* Testing framework – JUnit5
* Browser Automation – Selenium WebDriver
* Reporting – Allure framework

**Tasks**

* Automate 5 scenarios, which are described above
* Tests from one test class should be executed in one browser (if 3 test classes – browser should be opened 3 times, before each test class, not before each test method)
* Required patterns: Page Object (Page Factory), Singleton, Strategy (You can add more, if needed)
* If your project will contain DDT - store datasets in txt/xml/json files
* Project should be placed on GitHub or Bitbucket
* Tests should work in Chrome and Firefox
* Add switch in your code to run tests locally/ using Selenium Grid/SauceLabs/Docker (user can give params – url, port, etc.)
* If some test fails, attach screenshot, date and time, browser, platform version to your report
* Add cleanup