Добрый день!

Задание было следующим:

С помощью Selenium WebDriver+ Maven + Java + Allure + JUnit  автоматизировать следующий тестовый сценарий:

 1.       Открыть браузер FireFox  и развернуть на весь экран

 2.       Зайти на yandex.ru.

 3.       Перейти в яндекс маркет

 4.       Выбрать раздел Компьютеры// **БАГ спецификации, описал ниже**

 5.       Выбрать раздел Планшеты

 6.       Зайти в расширенный поиск // **БАГ спецификации,правльно “Перейти ко всем фильтрам”**

  7.       Задать параметр поиска от 20000 рублей.

 8.       Задать параметр поиска до 25000 рублей.

 9.       Выбрать производителей Acer и DELL

 10.   Нажать кнопку Применить.

 11.   Проверить, что элементов на странице 10// **БАГ спецификации, элементов 12**

 12.   Запомнить первый элемент в списке.

 13.   В поисковую строку ввести запомненное значение.

 14.   Найти и проверить, что наименование товара соответствует запомненному

 значению.

   Условия:

 ·         Необходимо параметризовать и вынести часто используемые/дублируемые инструкции в отдельные функции.

 ·         Предусмотреть использование recovery-действий, в случае неожиданного результата.

 ·         В каждом действии по переходу должны быть встроены логические проверки-ожидания открытия страницы.

 ·         Каркас –фреймворк unit-тестирования J-Unit.

 ·         Результатом выполнения должен быть Yandex.Allure отчет.

 ·         Проект должен быть оформлен как артефакт Maven.

 ·         Исходный код проекта должен быть выложен на github или bitbucket.​

**Структура вспомогательных страниц.**

В каждом классе страниц (MainPage, MarketPage, FilterPage) есть “база” вэб элементов, реализованная через HashMap. Так же есть методы находящие вэб элемент по ключу и выполняющие с ним действия (клик или ввод). Все это наследуется от класса AbstractPage, который содержит абстрактный метод наполняющий “базу” вэб элементами. Старался максимально унифицировать работу со страницами. Идея была такая - создал класс MainPage унаследовал его от AbstractPage , сразу переопределил абстрактный метод, наполнил базу вэб элементами(кнопкой яндекс маркет), создал метод по переходу в яндексмаркет ну итд.

**“Необходимо параметризовать и вынести часто используемые/дублируемые инструкции в отдельные функции.”**

Я написал параметраризованный тест, со следующими параметрами:

**минимальная цена, максимальная цена, бренд1, бренд2**.

Данные для первого теста **20000, 25000, Acer, DELL.**

Для второго **19000, 23000, ASUS, Lenovo.**

**Предусмотреть использование recovery-действий, в случае неожиданного результата.**

При переходе в яндекс маркет могут появиться 3 вида интерфейса каталога “Компьютеры”, раз на раз не приходится. Мой метод по переходу в каталог охватывает их все.

**Результатом выполнения должен быть Yandex.Allure отчет.**

Не успел до конца разобраться с Yandex.Allure. Смог только использовать комманду allure serve , и сохранить отчет как страницу. Allure report не сработала, а разбираться с плагинами ужне нет времени.