

## Пример Классификации тестов для приложения Wildberries

<https://www.wildberries.ru/>

1	Различные виды тестирования	
1.1	Функциональные тесты (Functional testing)	Проверяем основную функциональность приложения: <ol style="list-style-type: none"><li>1. Проверяем поле регистрации. Вводим корректные данные для регистрации. Проверяем, что регистрация проходит успешно.</li><li>2. Проверяем поле поиск. Проверяем, что поле поиск работает по валидным значениям, ищет по введенным значениям, и не ищет по не указанным.</li><li>3. Проверяем работу фильтров и сортировки на релевантность выдачи. Выбираем товар, устанавливаем фильтры. Проверяем, что сортировка по фильтрам отображает товар, соответствующий выбранным фильтрам, выборка релевантная.</li></ol>
1.2	Тесты производительности (Performance testing)	Определяем границы приемлемой производительности: - загрузка страницы 0,1-0,2 сек; - ответ\ обратная связь от действия пользователя (загрузка фото, ввод данных) – 0,5-1,0 сек. Проверяем изменение производительности: - при увеличении нагрузки; - при увеличении интенсивности выполнения операций. При отклонении от заданных параметров более чем на 20% принимаем решение об оптимизации производительности.
1.3	Нагрузочные тесты (Load testing)	Нагрузить функции приложения до предела, чтобы узнать до какой нагрузки сервис остается функционален. Например: <ol style="list-style-type: none"><li>1. Настроить Apache JMeter (или другой сервис) для проверки одновременной работы максимально возможного количества пользователей. - 10 запросов одновременно. Скорость ответа (с). % ошибок - 100 запросов одновременно. Скорость ответа (с). % ошибок - 10 000 запросов одновременно. Скорость ответа (с). % ошибок - 100 000 запросов одновременно. Скорость ответа (с). % ошибок</li></ol> Проверяем при какой максимальной нагрузке приложение работает без потери скорости, и ошибок.
1.4	Тестирование надёжности (Reliability testing)	Проведем проверку работоспособности нашего приложения при длительном (многочасовом, например

		<p>поставим сервер на неделю) тестировании со средним уровнем нагрузки (расчетная нагрузка по ТЗ+%). Проверим с самой обычной средней нагрузкой, не будет ли падений, неожиданного поведения, вызванного утечками памяти, переполнения базы, потери памяти. Т.е. проверяем стабильность работы приложения.</p>
1.5	Тестирование совместимости (Compatibility testing)	<ol style="list-style-type: none"> <li>1. Проверить запуск и работу приложения вб в разных браузерах: Chrome, Firefox, Opera, Edge, Safari</li> <li>2. Проверить запуск и работу приложения вб на разных ОС: Windows (разные версии), IOS (разные версии)</li> <li>3. Проверить запуск и работу приложения вб на разных устройствах. Использовать разные модели и конфигурации. Стационарный компьютер, Laptop, мобильный телефон, планшет.</li> <li>4. Проверить запуск и работу приложения вб с разных сетевых устройств, открыть на нескольких мониторах, разного разрешения.</li> </ol>
<b>2</b>	<b>Различные типы тестов</b>	
2.1	Позитивные (функциональные и нефункциональные) тесты	<ol style="list-style-type: none"> <li>1. Проверка строки поиск по валидным значениям: по слову, по подсказке, по фото, разными способами ввода, ввод символов, регистрозависимость, табуляция.</li> <li>2. Проверка регистрации с нормальным именем: ввод логина, пароля, получения смс.</li> <li>3. Выбор адреса и способа доставки из валидных значений</li> </ol>
		<ol style="list-style-type: none"> <li>1. Удобство использования приложения. Оценить интуитивность использования и графический дизайн.</li> <li>2. Проверить нет ли наложение текста в баннерах и сообщениях, выход за рамки области</li> <li>3. Защищенность. Войти в веб-приложение, используя действительные учетные данные. Выйти из приложения. Нажать кнопку НАЗАД браузера. Убедится, что вас попросили снова войти в систему или вы можете вернуться на страницу входа снова.</li> </ol>
2.2	Негативные (функциональные и нефункциональные) тесты	<ol style="list-style-type: none"> <li>1. Ввод недопустимых значений в поле регистрации</li> <li>2. Ввод максимально больших значений в строку поиска (более 300 символов)</li> <li>3. Загрузка недопустимых форматов файлов в поле поиск (более 8мб, pdf, csv, docx)</li> </ol>
		<ol style="list-style-type: none"> <li>1. Нагрузка. Проверить возможность работы приложения при одновременной работе максимально возможного количества пользователей.</li> <li>2. Производительность. Проверить скорость загрузки фото, с максимально допустимым размером</li> </ol>

		3. Защищенность. Тестирование приложения к различному виду инъекций (SQL, SOAP, LDAP, XPATN и т.д.)
<b>3</b>	<b>Различные форматы тестирования</b>	
3.1	Скриптовое тестирование (по тестам): для каких частей продукта и в каких случаях будет полезнее всего?	<p>Применяется в случаях:</p> <ul style="list-style-type: none"> <li>- При работе над большим проектом/ в большой команде</li> <li>- когда есть возможность планировать задачи, есть доступ к ТЗ и присутствует много документации</li> <li>- сложный продукт, много неочевидных моментов</li> <li>- достаточно времени и бюджета на проект</li> </ul> <p>Для тестирования бизнес-правил, безопасности. Тестирование функциональных полей приложения: поле ввода, поле регистрации, функции оплаты, функция сортировки и фильтра.</p>
3.2	Исследовательское тестирование (свободным поиском): для каких частей продукта и в каких случаях будет полезнее всего?	<p>Применяется в случаях:</p> <ul style="list-style-type: none"> <li>- когда нет доступа к ТЗ, нет возможности задать вопрос заказчику\разработчику</li> <li>- постоянный коллектив,</li> <li>- сотрудники с высокой квалификацией</li> <li>- чтобы обеспечить быструю обратную связь о новом продукте,</li> <li>- когда нужно быстро изучить продукт,</li> <li>- небольшой проект с маленькой командой.</li> </ul> <p>Для тестирования нашего приложения, не имея ТЗ, мы можем использовать туры Джеймса Уиттакера. Например, проведем «Тур супермодели», проверим <i>Юзабилити и Интерфейс</i>, узнаем о внешнем интерфейсе приложения: красив ли, привлекателен ли, правильно ли используются цвета, нет ли лишних артефактов в интерфейсе, нет ошибок и опечаток в названиях и баннерах, соответствует ли интерфейс принятым стандартам (сравниваем с конкурентами), ожиданиям пользователя. Так же полезным будет «Тур Коллекционера». Составим маинд карту приложения, и шаг за шагом будем проходить по всему <i>функционалу</i> приложения. Таким образом сможем протестировать все функциональные возможности приложения, чтобы примерно оценить тестовое покрытие, а также необходимость дополнительных исследований.</p>
3.3	Автоматизированное тестирование: для проверки какого функционала наиболее полезно? Почему именно для него?	<p>Применяется, когда тест подразумевается выполнение большого числа повторяющихся задач, содержит много вычислений и комбинаций.</p> <p>При <i>регрессионном тестировании</i>, так как тестировщики теряют концентрацию и начинают пропускать дефекты. Для проведения <i>нагрузочного</i> тестирования и <i>производительности</i>. Позволяет учесть все возможные комбинации взаимодействия пользователя с интерфейсом в веб-браузерах и прочих системах. Например:</p>

		<ul style="list-style-type: none"> <li>- тестирование «поля поиск» на производительность, при одновременном использовании несколькими пользователями, замеряем скорость работы поиска.</li> <li>- на работу функции «скидка», когда нам надо перебрать и проверить большое количество вариантов и комбинаций.</li> </ul>
--	--	--