**Beet Seed**

**1.**

|  | **Функциональные** | **Нефункциональные** | **Связанные с изменениями** |
| --- | --- | --- | --- |
| **Что проверяется** | Проводиться проверка функциональности системы (функциональная пригодность) и функциональные требования к системе. Также здесь могут выполняться проверки пользовательского интерфейса и взаимодействия систем внутри программы. | Все виды тестирования производительности (производительность, нагрузочное, стресовое, стабильности или надежности, объемное тестирование), удобство использования, удобство поддержки, портативности, установки, отказ и восстановление, конфигураций и безопастности. | Смоук необходим для быстрой (поверхностной) проверки работаспособности системы после внесений изменений в код (применимо к куску функционала);  Регрессия нужна для проверки всей системы целиком, после добавления нового функционала или внесения изменений в старый;  Санити проверяет какой-то конкретный функционал “в глубину”.  Ре-тестинг проверяет исправленный функциона и что он работает правильно.  Тестирование сборки проверяет и даёт быструю оценку готовности всей сборки (применимо ко всей сборке в целом). |
| **Когда применяются** | Применяется на протяжении всего цикла разработки ПО. Особенно важно проводить функциональное тестирование после внесения изменений в код, перед релизом и при интеграции с другими системами. Но обычно применяются на этапе системного тестирования. | Тестирование проводится на протяжении всего жизненного цикла разработки, особенно актуально перед релизом и при интеграции с другими системами. Обычно применяется на этапе системного и приёмочного тестирования. | Применяется как на этапе системного тестирования так и с процессе приёмочного. Зачастую данный вид тестирования применяется если в рабочую систему были внесены или вносятся любые изменения для определения и подтверждения корректности работы системы. |
| **Ограничения** | Ограничений касательно функционального тестирования мало, но всё-таки могу выделить следующие: неполное покрытие, в связи с тем, что покрыть всё невозможно, есть вероятность пропуска важных проверок; зависимость от качества тестовых данных; трудоемкость; возможностью упущения логических ошибок ПО; вероятностью избыточного тестирования | Проведение нефункционального тестирования касательно некоторых его видов осуществимы только на готовом продукте (например удоство использования или портативность и т.д.).  Также ограничения в виде: субъективности человека, который проводит тестирование; высокой стоимости и сложности автоматизации; взаимозависимостей характеристик и зависимостей от окружения | Данный виде тестирования не может быть проведён на ранних этапах разработки. Под этим имееться в виду этапы планирования, анализа и подготовки проектной документации. Также, сюда можно затратностью по времени (особенно по отношению к регрессионному тестированию), сложностью изоляции дефектов, неполное покрытие тестами, также постоянные изенения требований. |
| **Особенности** | Особенностью является проверка реализуемости функциональных требований приложения, т.е. способность приложения в заданных критериях решать возложенные на него (на приложение) задачи.  Отвечает на вопрос: Что делает система? | Описывают тесты, необходимые для определения характеристик ПО, которые могут быть измерены различными величинами.  Отвечает на вопрос: Насколько хорошо работает система? | После проведения необходимых изменений, таких как исправление бага/дефекта, внедрение новой функциональности, рефакторинга, программное обеспечение должно быть перетестировано для подтверждения того факта, что проблема была действительно решена и/или старая часть приложения работает без изменений. |

**2. Регрессионное тестирование**  — это такой вид тестирования, который проводиться после внесения или добавления новых изменений в существующий код с целью проверки и удостоверения в том, что изменения не нарушили целостность готовой системы и не повлекли за собой проблем.

**Ре-тестинг** — это конкретный вид тестирования, который направлен на проверку/ перепроверку исправленого функционала. То есть, проводиться тестировщиком после выполненных работ программистом.

Разница заключается в том, что ре-тестинг проводиться “точечно” после исправления проблем в конкретном функционале и проверяет только его (то что исправилось). В то время как регрессия, выполняется для убеждения в том, что данный функционал, который был исправлен и “внедрён” в существующий код/ систему не нарушил её целостность и работоспособность.

**Beet Sprout**

**1.** Всё зависит от конктекста и в теории функциональное тестирование может быть проведено без нефункцонильного при определённых ограничениях на проекте, то есть это возможно.

Но я не сторонник такого варианта и считаю, что если тестировать качественно, то необходимо покрывать, по крайней мере стараться покрыть продукт всеми видами тестирования.

Почему это важно? Как функциональный так и нефункциональный вид тестирования имеет ряд целей который он предследует, а именно позволяет нам измерить такие показатели как удовлетворённостью системы по средствам измерения величин. И позволяет нам понять, насколько хорошо работает система. Так как мы разрабатываем систему для людей/пользователей, для бизнеса/владельца продукта важно получать пользу/выгоду от продукта, а если продукт будет выполнять свои функции, но не будет предоставлять удобство и комфорт во время использования, то результат может быть отрицательным, что в свою очередь принесёт потери и скажеться на прибыли.

**2. Смоук тестирование —** это начальный этап тестирования программного обеспечения, который позволяет быстро оценить, готова продукт к дальнейшему тестированию. По сути, это "дымовая проверка", позволяющая выявить проблемные места на базовом уровне, которые не позволят проводить дальнейшие (углублённые) проверки.

Каждый раз, как мы берём на проверку какой-то кусок кода или функционал нам стоит выполнить “беглую” проверку на предмет верификации и валидации, чтобы иметь понимание на тему того, можем ли мы проводить полноценное тестирование. Данный подвид тестирования необходим, чтобы повысить эффективность нашей работы.

Как я указывал ранее всё зависит от контеста и ситуации на проекте. Есть ситуации, когда у тестировщик на проекте мало времени, чтобы провести полный тест функционала или могут возникнуть какие-либо непредвиденные проблемы, которые не позволят ему сделать это. Тогда тестировщик может воспользоваться данным подвидом и в кратчайший срок дать свою оценку работаспособности и качества функционала.

В целом, смоук тестирование позволяет быстро оценить качество новой сборки и обеспечить стабильность системы. Рекомендуется включать смоук тестирование в каждый цикл разработки, чтобы повысить качество продукта и сократить время на его создание.

**Mighty Beet**

**1.**

| **ID** | Valid\_#0001 | | | |
| --- | --- | --- | --- | --- |
| **Priority** | 1 (High) | | | |
| **Summary** | Checking the cat’s foto uploading as a logged-in user from the “Add news” modal | | | |
| **Pre-condition** | Move to the https://meu-meu.com resource; A user is logged into the system | | | |
|  | **Steps** | **Test Data** | **E.R.** | **Comments** |
| **1** | Tap on the “Add news” button in the top right corner. |  | The “Add news” modal is opened. |  |
| **2** | Fill in the “News” section | Max length = 150 symbols |  |  |
| **3** | Tap on the “Add foto” button and select cat’s foto from your gallery | A foto size should not exceed 5 mb; only one foto per time |  |  |
| **E.R.** | The green “Succeed” pop-up is displayed below the “Add foto” section. A “Publish” button is active. | | | |
| **Post-condition** |  | | | |

| **ID** | Invalid\_#0001 | | | |
| --- | --- | --- | --- | --- |
| **Priority** | 2 (Medium) | | | |
| **Summary** | Checking the dog’s foto uploading as a logged-in user from the “Add news” modal | | | |
| **Pre-condition** | Move to the https://meu-meu.com resource; A user is logged into the system | | | |
|  | **Steps** | **Test Data** | **E.R.** | **Comments** |
| **1** | Tap on the “Add news” button in the top right corner. |  | The “Add news” modal is opened. |  |
| **2** | Fill in the “News” section | Max length = 150 symbols |  |  |
| **3** | Tap on the “Add foto” button and select dog’s foto from your gallery | A foto size should not exceed 5 mb; only one foto per time |  |  |
| **E.R.** | The red “You can add only cat’s foto” pop-up is displayed below the “Add foto” section. A “Publish” button is blocked. | | | |
| **Post-condition** |  | | | |

| **ID** | Invalid\_#0002 | | | |
| --- | --- | --- | --- | --- |
| **Priority** | 2 (Medium) | | | |
| **Summary** | Checking the human’s foto uploading as a logged-in user from the “Add news” modal | | | |
| **Pre-condition** | Move to the https://meu-meu.com resource; A user is logged into the system | | | |
|  | **Steps** | **Test Data** | **E.R.** | **Comments** |
| **1** | Tap on the “Add news” button in the top right corner. |  | The “Add news” modal is opened. |  |
| **2** | Fill in the “News” section | Max length = 150 symbols |  |  |
| **3** | Tap on the “Add foto” button and select human’s foto from your gallery | A foto size should not exceed 5 mb; only one foto per time |  |  |
| **E.R.** | The red “You can add only cat’s foto” pop-up is displayed below the “Add foto” section. A “Publish” button is blocked. | | | |
| **Post-condition** |  | | | |

| **ID** | Valid\_#0002 | | | |
| --- | --- | --- | --- | --- |
| **Priority** | 1 (High) | | | |
| **Summary** | Adding friends to the account from the “Friends” page as a logged-in user | | | |
| **Pre-condition** | Move to the https://meu-meu.com resource; A user is logged into the system | | | |
|  | **Steps** | **Test Data** | **E.R.** | **Comments** |
| **1** | Tap on the “Friends” button from the side menu. |  | The “Friends” page is opened. |  |
| **2** | Select a friend from the list | A “Little cat Sarah” account |  |  |
| **3** | Tap on the “+” symbol next to the user’s foto |  |  |  |
| **E.R.** | The “‘Little cat Sarah’ user is added to your friend list” pop-up is displayed in the top middle of the screen. A user is added to the “Friends” page. | | | |
| **Post-condition** |  | | | |

| **ID** | Valid\_#0003 | | | |
| --- | --- | --- | --- | --- |
| **Priority** | 1 (High) | | | |
| **Summary** | Adding “Likes” for the posts as a logged-in user on the “News” page | | | |
| **Pre-condition** | Move to the https://meu-meu.com resource; A user is logged into the system | | | |
|  | **Steps** | **Test Data** | **E.R.** | **Comments** |
| **1** | Tap on the “News” button from the side menu. |  | The “News” page is opened. |  |
| **2** | Select the first news | “Our walking” news |  |  |
| **3** | Tap on the “👍🏼” symbol at the bottom right of the selected news |  |  |  |
| **E.R.** | The “👍🏼” symbol is added to the news. The counter is counted and displays amount of likes. | | | |
| **Post-condition** |  | | | |

**2.** Стоит проверить удобство использования нашего приложения. Считаю данную проверку важной, так как нашим пользователям должно быть комфортно пользоваться приложением. Важно, чтобы все элементы были “под рукой”, меню было простым и не сложным, структура приложения должна иметь интуитивно понятный дизайн. Так как данное требование является субъективным, проверка будет происходить на основании дизайна, который был разработан командой дизайнера в тесном сотрудничестве с заказщиком.  
 Также необходимо провести тестирование производительности, то есть масштабируемость системы под нагрузкой. Проверять будем под нагрузкой до 40%. Данная проверка должна показать нам время ответа от сервера, насколько быстро пользователю будет отрисовывается информация. Также эта проверка поможет нам проверить как быстро загружаются наши разделы/страницы приложения.

Стоит проверить безопастность нашего приложения, то есть насколько хорошо наше приложение устойчиво к взломам и насколько надёжно защищены конфиденциальные данные наших пользователей. В данном подвиде нефункционального требования стоит проверить валидацию наших полей в момент регистрации и в момент логина. Стоит добавить опредёлнные требования к паролю (мин кол-во символов и спец знаков) и систему смены пароля каждые 6 месяцев.