**Test-plan**

**Test-plan** - планировка того, как мы собираемся проводить работу по тестированию (напр. что мы тестируем, какие техники мы будем применять) сам по себе, test-plan пишется согласно стандартизации технической документации IEEE829 и имеет 15 пунктов. Для каждого релиза, тест-план пишется свой.

В настоящее время, написание test-plan постепенно теряет свою популярность, так как очень сложно увидеть на перед весь релиз и расписать, что будет протестировано, а что нет

1. **Introduction** (Вступление) - краткая сводка того, что содержит наш test-plan. (напр. Этот документ является тест-планом проекта "мобильный банк", в нем описано то, что мы планируем протестировать и то, что мы не планируем тестить, что нужно команде для успешного проведения тестирования, какие методы тестирования мы планируем применять, какие критерии качества для нас приемлемы и т.д.)

2. **Features to be tested** (тестирование функционала):

* Launch (запуск приложение)
* Конвертация валюты
* Online-помощник (типа чат-бот)

3. **Features no to be tested** (функционал, который мы не будем тестировать):

* Мы не собираемся тестировать отображение банкоматов на карте
* Мы не собираемся тестировать функционал QR-code
* Мы не собираемся тестировать функцию управления финансами

4. **Test-items** (тестирование единицы) - нечто иное как декомпозиция, мы делим один функционал на мелкие составляющие, в примере ниже речь идет о функции face-id

* работает ли у нас face-id в ночное время
* работает ли у нас face-id если смартфон повернут горизонтально
* предлагает ли наш смартфон пройти face-id сразу при открытии приложения

5. **Test approaches** (Тестовый подход/стратегия) - мы описываем каким способом мы хотим проводить тестирование, какой вид тестирования мы будем использовать. (напр. Мы будем проводить ручное тестирование с использованием black-box. Для выбора тестовых значений мы будем применять технику эквивалентных классов, для создания тестового сценария, мы будем использовать технику тестирования состояния и переходов (State and Transitions Tech))

6. **Pass/Fail item criteria** (Acceptance criteria/Критерии принятия продукта) - Pass/Fail item criteria это два подхода для к критерии приемки, ее можно применять на весь продукт целиком и тогда мы в конце делаем приемочное тестирование прогоняем все наши тесты для этого этапа и если мы соответствуем заранее установленной критерии то мы уходим в релиз, если нет то мы ждем доработок, новый RC Build и делаем все по новой. Так же, мы можем применять Acceptance criteria к каждому item.

7. **Suspension/Resumpton criteria** (Критерия приостановки/продолжения тестирования) - заранее прописанная критерия, при которой тестировщики могут прекратить работу. (напр. если приложение не проходит Smoke тест успешно, мы остановим процесс тестирования)

8. **Tester/task** (Задача тестировщика) — это пункт из серии, что должен делать тестировщик (напр. тестировщики будут писать тестовую документация и отправлять тест-репорты. Дабы разгрузить работу разработчика, тестировщики должны будут писать unit/IT тесты)

9. **Environmental needs** (Потребность в окружение) - описание того, что нужно тестировщику для написания тестов (напр. для успешного тестирования мобильного приложения, тестировщику понадобиться смартфоны с OS android в количестве 15 штук)

10. **Staff and training** (Персонал и обучение) - потребность в особом персонале и тренинге для него (напр. я являюсь специалистом ручного тестирования, чтобы я мог заниматься и автоматизированным тестированием, мне не обходимо пройти какие-то курсы)

11. **Test deliverables** (что мы предоставим по окончанию тестирования) - по окончанию тестирования, мы предоставим заказчику развернутый документ, в котором будет описано, в каком окружении были проведены тесты и в каком окружении было найдено больше дефектов

12. **Schedule** (График) - график нашей работы. Когда и какое действие, команда разработчиков будет делать

|  |  |
| --- | --- |
| Написание тестовой документации | 10 августа – 20 августа |
| Проведение Smoke тестов | 4 ноября – 8 ноября |
| Анализ требований | 1 августа – 9 августа |

13. **Risk** - описание того, что делать если у нас что-то пойдет не так. (напр. если к 1 ноября мы не получим RC build, то есть большая вероятность, что мы не сможем завершить тестирование в срок)

14. **Responsibility** (кто-за что отвечает) это тоже график, в котором описано, кто отвечает за какую-то работу + его роль. Делается это для того, чтобы заказчик знал к кому обратиться.

|  |  |  |
| --- | --- | --- |
| Иванов | Отвечает за приемочное тестирование | QA Manual middle |
| Петров | Отвечает за написание документации | Team Lead |
| Сидоров | Отвечает за ручное тестирование | QA Manual |

15. **Approvals** (Утвердившие) – как правило написано по шаблону (Фамилия, позиция, дата, подпись). Пункт “Утвердившие” нужен для того, чтобы в тех случаях, когда у нас что-то пошло не так и нам зададут вопрос, почему вы не протестировали продукт с последней версии Ubuntu, мы могла сказать: “В Тест плане было написано, что мы не собираемся тестировать продукт в последней версии Ubuntu и это утвердил ответственный человек”