

# Планирование тестовых испытаний

---

# Введение

- В небольших проектах тестировщики не углубляются в вопросы качества, не стараются предположить проектные риски, не заботятся о критериях качества и тд.
- Тестировать нужно не только разные версии программы. Приступать к тестированию следует уже на этапе сбора требований. И тестировать далее на последующих этапах вплоть до выхода продукта. Частенько продукт продолжают тестировать и на этапе поддержки.

# Связь планирования тестовых испытаний с жизненным циклом ПО

- Есть много взглядов на жизненный цикл ПО. В контексте тестирования и планирования тестовых испытаний проще всего рассматривать следующую упрощённую модель:
  - Начало (inception).
  - Уточнение (elaboration).
  - Разработка (construction).
  - Передача заказчику (transition).



# Жизненный цикл ПО

- 1. Начало
  - А) сбор требований
    - собраны пожелания заказчика ->
    - сформулированы бизнес-требования ->
    - написаны функциональные требования;
  - Б) анализ требований для создания архитектуры и дизайна
    - как только появляются первые документы – тестировщики начинают их анализировать и готовиться к тестированию;
    - чуть позже разработчики начинают работу



# Жизненный цикл ПО

## 2. Уточнение

### А) разработчики

- пишут отдельные модули и юнит-тесты

### Б) тестировщики

- Проводят модульное тестирование и интеграционные тесты

### В) обе группы специалистов

- активно уточняют/дорабатывают требования и дизайн



# Жизненный цикл ПО

## 3. Разработка

### А) программисты

- пишут главные функции продукта (пиковая активность)

### Б) тестировщики

- тестируют функциональность на всех уровнях тестирования (пиковая активность в конце фазы)

### В) работа с требованиями

- сокращается, в конце фазы уже невозможно внести серьезные изменения



# Жизненный цикл ПО

## 4. Передача заказчику

### А) команда поддержки

- развертывает систему у заказчика

### Б) сторонние тестировщики

- проводят приемочное тестирование

### В) тестировщики/программисты

- активность спадает



# Области ответственности тестировщиков:

- Планирование тестов
  - разработка методологии и плана тестирования
  - участие в принятии стандарта качества
  - разработка спецификаций тестов
- Разработка и выполнение тестов
  - создание ручных и автоматизированных тестов
  - выполнение тестов
  - управление билдами (оценка состояния проекта)
- Отчеты по тестам
  - сообщить проектной группе о качестве продукта
  - отслеживать состояние дефектов





# Тестовый план

- **Тест план (Test Plan)** – это документ, описывающий весь объем работ по тестированию, начиная с описания объекта, стратегии, расписания, критериев начала и окончания тестирования, до необходимого в процессе работы оборудования, специальных знаний, а также оценки рисков с вариантами их разрешения



# Планирование и тестовый план

Тестовый план – документ проектной документации, который описывает:

- процесс тестирования конкретного продукта в конкретном проекте;
- что, когда, кем и как будет тестироваться;
- компоненты тестирования;
- команду тестировщиков;
- стратегию и методы тестирования;
- критерии качества и риски тестирования;
- график работ.



# Тест план

- Шаблоны:
  - [Test Plan Template RUP](#)
  - [Test Plan Template IEEE 829](#)



# Необходимые действия на стадии планирования

- Понять, что за продукт будет тестироваться, как он работает и для чего он нужен
- Понять, как продукт будет использоваться
- Хорошо изучить требования к продукту
- Решить какие части продукта будут тестироваться и как
- Убедиться, что выбранные области в полном объеме покрыты тестами
- Решить, какие из методов и техник тестирования наиболее эффективны для проверки нашего продукта
- Определить критерии качества продукта
- Определить и приоритизировать риски, то есть ситуации, которые приведут к ухудшению качества программного продукта. А также подумать о том, как предотвратить возникновение подобных ситуаций, и как из них выйти
- Выводы по всем вышеперечисленным пунктам попадают в тест-план, который нужно утвердить и разослать всем заинтересованным лицам
- Создать такое тестовое окружение, которое будет максимально приближено к условиям, в которых продукт будет эксплуатироваться (production environment)



# Артефакты, создаваемые на стадии планирования

К артефактам, создаваемым на стадии планирования можно отнести:

- тестовый план;
- матрица конфигураций, которая может быть включена в тестовый план как отдельный раздел;
- запрос на выделение тестового оборудования.



# Сложности планирования

- К сожалению невозможно всё предусмотреть и всё запланировать. Всегда будут возникать какие-то сложности или непредусмотренные ситуации во время работы. Будут возникать ситуации, когда будет трудно расставить приоритеты той или иной деятельности: например, какие тесты выполнить в первую очередь, а какие выполнять необязательно, (или выполнить, если останется время); какие виды тестирования более приоритетны, а какие – менее; какая функциональность более важна, а какая – менее.
- Также будут возникать различные ограничения, которые невозможно контролировать или на которые невозможно повлиять.
- Однако, всегда можно попросить помощи у менеджера, заказчика, группы разработчиков. Также, если видно, что что-то будет мешать тестированию, следует проинформировать об этом менеджера и заказчика и добиться понимания ситуации от них, а также согласовать действия, которые будут предприняты в данной ситуации.
- **Пример.** В процессе подготовки очередного билда программисты столкнулись с трудностями и выпустили билд на два дня позже запланированного срока. В результате график тестирования тоже сместился и у тестировщиков будет меньше времени, чтобы выполнить свою работу. Следует согласовать заранее свои действия в этом случае.

# Риски

- **Риск** – сочетание вероятности наступления события и последствий, вызванных этим событием.
- Думая о рисках, следует думать о том, какие риски могут возникнуть на проектном уровне. Эти риски на первый взгляд могут не касаться тестирования, но на самом деле их последствия могут напрямую влиять на работу команды тестировщиков.

**Пример.** В продукте разрабатывается несколько приложений. Для того, чтобы протестировать функциональность приложения А, требуется использовать функциональность приложения В. Разработчики запланировали реализовать сначала функциональность приложения А, а только затем приложения В.



# Секции тестового плана

Секции тестового плана включают в себя:

- Перечень работ
- Критерии качества и оценка качества процесса
- Оценка рисков
- Документация и письма
- Тестовая стратегия
- Ресурсы
- Метрики
- Расписание
- Ключевые точки





# Перечень работ (scope of work)

- Сюда включается перечень функциональных областей приложений, которые будут подвергаться тестированию. Здесь же может быть перечень компонентов или функциональности, которые не будут тестироваться по тем или иным причинам. Например, эту функциональность реализует другая фирма. Или в проекте используются какие-то уже готовые компоненты. Если есть какие-либо ограничения тестирования, их тоже можно здесь перечислить.



# Пример

- В приложении используется визуальный HTML-редактор, который разработан другой фирмой. Ограничение тестирования состоит в том, что мы не собираемся тестировать функциональность самого редактора, поскольку предполагаем, что это сделала компания-разработчик данного редактора. Мы сосредоточимся на тестировании взаимодействия редактора с нашим приложением. Другими словами, будем проводить интеграционное тестирование.

Однако, если в процессе интеграционного тестирования обнаружатся какие-либо дефекты в самом редакторе, их следует документировать с соответствующей пометкой, чтобы потом наши разработчики могли разобраться, в действительности ли данные дефекты являются дефектами самого редактора, и, если так, мы сообщим о них компании-производителю данного компонента для последующего исправления.

# Критерии качества и оценка качества процесса (quality criteria and process assessment)

Здесь отражается перечень критериев качества, на основании которых будет приниматься заключение об уровне качества продукта и возможности передачи продукта заказчику. Критерии качества относятся к качеству продукта. В тестовых планах могут быть записаны и критерии качества самого процесса тестирования (и разработки), с целью последующей оценки качества процесса. Это необходимо, чтобы в случае необходимости существовала возможность оценить, насколько грамотно был построен процесс тестирования, были ли какие-либо проблемы, и, если были, разобраться, почему, а также выработать рекомендации по их предотвращению в будущем.

# Оценка рисков (risk assessment)

Здесь речь идёт как раз о тех самых рисках (негативных ситуациях), которые могут возникнуть в процессе работы над проектом и помешать (негативно повлиять) на тестирование продукта. Кроме описания самого риска, следует указать вероятность его возникновения и степень влияния на проект. Производная этих двух величин даёт показатель, который может рассматриваться как **степень серьёзности риска**. Чем выше этот показатель, тем больше внимания нужно уделить этому риску: обдумыванию последствий, составлению плана его предотвращения или выхода из ситуации, если риск наступит. Если риск случился (т.е. ситуация наступила), возникает реальная проблема («issue», «hot issue»), которую необходимо решить.



# Документация и письма (test documentation and deliverables)

- В этой секции размещается перечень артефактов (результатов деятельности). Например:
  - тест план;
  - тестовые сценарии;
  - тестовые автоматические скрипты;
  - дефект-репорты;
  - отчеты о результатах тестирования.
- Также указывается, кто будет высылать данный артефакт, как часто, каким способом, кому и т.д.



# Тестовая стратегия (test strategy)

- Самый большой и один из самых важных разделов плана.
- Здесь расписывается стратегия тестирования, методы и типы тестов, каким образом будет выполняться тестирование, почему именно так и т. д.
- Конкретное содержание этого раздела зависит от фирмы-разработчика, проекта, заказчика и т. д.



# Тестовая стратегия (test strategy)

Этот раздел может включать подразделы:

- Критерии приёмки билдов
- Методы тестирования
- Типы тестирования
- Уровни тестирования
- Отслеживание ошибок
- Использование метрик



# Ресурсы (resources)

- **Человеческие ресурсы:** перечень ключевых людей на проекте (менеджер проекта, представители заказчика, лидер команды разработчиков и т.д.), список тестировщиков с их ролями на проекте, а также с зонами ответственности.
- **Аппаратные ресурсы (hardware):** сюда входит перечень тестовых серверов и рабочих станций, инструментов, используемых для тестирования или для вспомогательных работ, описание тестового окружения.
- **Программные ресурсы (software):** операционные системы, СУБД, серверы приложений, веб-серверы и т.д.





# Метрики (metrics)

- **Метрика** – это числовая характеристика, позволяющая оценить тот или иной аспект программного продукта или процесса в целом. **Например:** количество дефектов, найденных за неделю в тех или иных областях продукта. Эта метрика позволяет оценить, как много дефектов в той или иной функциональности. Что, в свою очередь может позволить сделать немало выводов или, например, подтолкнуть к разбору причин такого количества багов.



# Правило

- **Правило метрики:** Билд считается неприемлимым, если в нем есть хотя бы один Critical или High баг, либо 5% функционала не протестировано или имеет Medium или Low баги.



# Расписание и ключевые точки (schedule and milestones)

- В этой секции описывается график тестирования в согласовании с графиком выпуска билдов и проектным планом, который разрабатывается менеджером проекта.
- Сюда же включаются основные даты (milestones): например, даты окончания промежуточных фаз работы над проектом.
- График тестирования нужен для того, чтобы чётко понимать, когда и что следует делать, ничего не пропустить, ничего не забыть и т.д. Он же упрощает контроль за ходом работ по тестированию, а также позволяет оценить текущую ситуацию, определить, всё ли выполнено из того, что было запланировано.



# Критерии хорошего тестового плана

- Тест план должен быть:
  - **Полным**
  - **Корректным**
  - **Недвусмысленным.**



# Критерии хорошего тестового плана

- В тест плане должны быть:
  - определены цели тестирования, тестовый подход, стратегия, методы, виды тестирования. Запланированный подход должен быть реально выполним
  - установлены реалистичные критерии качества.
  - определены критерии прохождения приёмочного теста и условия прекращения тестирования.
  - определены все артефакты, подлежащие сдаче, поставке или распространению (заказчику, проекту и т.д.)
  - перечислены тестовые ресурсы (люди, оборудование) с указанием ролей, назначений, ответственности.



# Критерии хорошего тестового плана

- Также:
  - Должно быть определено и описано тестовое оборудование, окружение, программное обеспечение.
  - Должен быть определён график тестирования, он должен быть реалистичен и выполним.
  - Тест-план должен соответствовать принятому в компании шаблону, если на проекте не решено иначе: например, использовать шаблон заказчика.



# Преимущества хорошего тестового плана

Давайте подумаем



# Преимущества хорошего тестового плана

- В условиях постоянного ограничения и нехватки времени, хорошо распланированный, систематизированный подход позволяет достичь лучших результатов работы, а также позволяет обнаруживать большее количество ошибок, чем неорганизованная, плохо распланированная деятельность.
- Тестовый план позволяет управлять процессом тестирования более эффективно.
- Тестовый план позволяет увидеть и понять минимальный уровень тестирования и получить представление об уровне проводимого тестирования каждой области продукта.
- Тестовый план позволяет достичь соглашения между исполнителями, заказчиком и менеджером, о том, каким образом и в какие сроки будет проводиться тестирование.





# Пример тестового плана

# Тест план. Закрепляем

- **Что надо тестировать?**
  - описание объекта тестирования: системы, приложения, оборудования
- **Что будете тестировать?**
  - список функций и описание тестируемой системы и её компонент в отдельности
- **Как будете тестировать?**
  - стратегия тестирования, а именно: виды тестирования и их применение по отношению к объекту тестирования
- **Когда будете тестировать?**
  - последовательность проведения работ: подготовка (Test Preparation), тестирование (Testing), анализ результатов (Test Result Analysis) в разрезе запланированных фаз разработки
- **Критерии начала тестирования:**
  - готовность тестовой платформы (тестового стенда)
  - законченность разработки требуемого функционала
  - наличие всей необходимой документации
  - ...
- **Критерии окончания тестирования:**
  - результаты тестирования удовлетворяют критериям качества продукта:
    - требования к количеству открытых багов выполнены
    - выдержка определенного периода без изменения исходного кода приложения Code Freeze (CF)
    - выдержка определенного периода без открытия новых багов Zero Bug Bounce (ZBB)
  - ...
- **Риски и управление ими**
  - поздняя поставка ПО
  - перебои в работе сервисов третьей стороны
  - ...