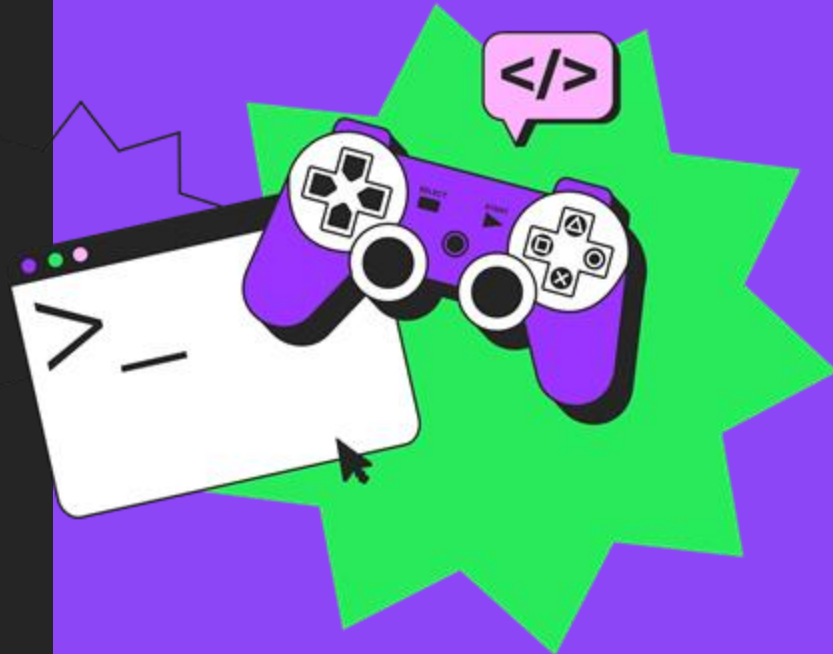


# Bootcamp Project Manager

Семинар 18. Что нужно для завершения проекта





## План

1. Вводный урок
2. С чего начать свой проект
3. Как идентифицировать заинтересованные стороны проекта
4. Бизнес-функциональные требования, Техническое задание и Product Vision
5. Что такое содержание проекта и содержание продукта
6. Что такое дорожная карта проекта
7. Как сформировать иерархическую структуру работ
8. Как построить план-график своего проекта
9. Как идентифицировать и оценить риски проекта
10. Как определить команду проекта
11. Что такое Устав проекта
12. Как начать реализацию проекта
13. Традиционные методологии разработки
14. Итеративная разработка
15. Что делать, если сроки проекта близки к провалу
16. Как комбинировать различные методологии управления разработкой продукта
17. Управление командой проекта
18. **Что нужно для завершения проекта**
19. Как проанализировать свой проект при его завершении
20. Как извлечь опыт из своего проекта



## План – Учебный проект

21. Учебный проект. Идентификация и оценка стейкхолдеров
22. Учебный проект. Содержание проекта
23. Учебный проект. Дорожная карта проекта
24. Учебный проект. План-график проекта
25. Учебный проект. Риски проекта
26. Учебный проект. Команда проекта
27. Учебный проект. Устав проекта
28. Учебный проект. Организация работы команды
29. Защита проекта





## Что будет на семинаре сегодня

- ✚ Приёмочное тестирование для сдачи проекта
- ✚ Общие мысли о тестировании
- ✚ Типы тестов при завершении проекта
- ✚ Формирование тест-планов
- ✚ Эксплуатационные испытания продукта проекта
- ✚ Кейсы



Вопрос

Зачем нужно тестирование продукта проекта?





# Финальное/приёмочное тестирование

**Цель финального тестирования (эксплуатационного испытания)** – выявить недостатки разработанного продукта или его части и устранить их, чтобы полученные технические и пользовательские показатели продукта совпадали с запланированными в самом начале проекта.



***UAT (User Acceptance Testing)***

***Пользовательское приёмочное тестирование***



Вопрос

Что делает РМ при тестировании?





# Роль РМ

Правильное выстраивание **процессов тестирования.**







# А на самом деле?

**Цель финального тестирования (эксплуатационного испытания)** – убедить клиента подписать акт приёма-передачи.





Вопрос

Какие бывают сценарии развития событий при  
тестировании продукта?





# Процесс приёмки проекта

## Плохой сценарий:

1. Заканчиваем разработку
2. Заказчик делает приёмку тем способом, который сам выберет
3. Мы лихорадочно делаем правки, от части правок пытаемся уклониться
4. Манипуляции и давление на клиента пока он не сдастся и не подпишет акт приёмки





# Процесс приемки проекта

## Хороший сценарий:

1. При планировании бьём проект на фазы
2. При планировании проекта пишем также тест план, в котором определяемся верхнеуровнево как будет проходить приёмка и метрики здоровья продукта
3. Согласовываем это с заказчиком
4. В ходе работ над фазой пишем планы и инструкции для UAT-тестирования
5. Делаем официальную приёмку в конце каждой фазы





# Процесс приемки проекта

## Экстремальный сценарий (или продуктовый):

1. При планировании бизнес-фичи пишем тест план, сценарии приёмки и ключевые метрики
2. В ходе работ над фичей пишем план и инструкции для её UAT-тестирования
3. Делаем UAT для каждой фичи по её завершению



Как оформлять приемку фичи? - договоренность о специальной колонке на доске Jira



Вопрос

Это всё хорошо.  
Но что делать если всё пошло по плохому сценарию?





# Что делать если всё пошло по плохому сценарию?

Главная цель “зафиксироваться”.

Получить ФИНАЛЬНЫЙ список правок. И на его основание написать и согласовать ФИНАЛЬНЫЙ план.

Подтвердить с заказчиком, что выполнение этих пунктов приведёт к сдаче проекта.

Если это не помогает - задумайтесь о фиксировании убытков (не попали в цели либо проблема на стороне клиента)





# Процессы тестирования

**Ошибка РМ** - не видеть процесс, который есть

- подготовка задач к передаче в тестирование
- передача
- приоритезация
- подготовка сред тестирования
- и тд. (процессы могут быть разными)





Вопрос

Какие Вы знаете типы приёмочного тестирования?





# Еще одно название приёмочного тестирования

**UAT (User Acceptance Testing) = Пользовательское приёмочное тестирование**

## Цели:

- ✓ Определить, работает ли продукт так, как задумывалось при его создании.
- ✓ Определить, явно ли обозначены все доступные функции.
- ✓ Проверить продукт на наличие багов и сбоев, которые мешают ему выполнять свои основные функции.

**Важно:** выявить кто должен проводить UAT,  
установить сроки и внести их в план проекта





# Типы тестирования UAT

- ✓ Альфа/бета тестирование
- ✓ Контрактное приёмочное тестирование
- ✓ Законодательное приёмочное тестирование
- ✓ Операционное приёмочное тестирование
- ✓ Тестирование по стратегии чёрного ящика





# Альфа/бета тестирование

- **Альфа-тест:** роль пользователей продукта возлагается на штатных сотрудников и члены проектной команды (в частности, на разработчиков).
- **Бета-тест:** проводится уже реальными, специально отобранными пользователями.





# Контрактное приёмочное тестирование (contractual acceptance testing)

**Цель:** проверить, соответствует ли разработанный продукт проекта контрактным требованиям, согласованным всеми заинтересованными сторонами.

**Используется** для того, чтобы убедиться в том, что сторонняя команда разработчиков выполнила свои договорные обязательства.





# Законодательное приёмочное тестирование (regulation acceptance testing)

**Показывает**, что продукт соответствует всем законам и предписаниям своей отрасли и юрисдикции.

Такое тестирование необходимо проводить в сферах здравоохранения и финансов.





# Операционное приёмочное тестирование (operational acceptance testing)

## Цель:

убедиться, что система выполняет свою роль в среде эксплуатации согласно бизнес-модели.

**Позволяет оценить** следующие процессы:

- ✓ онбординг (эффективное включение пользователей в работу с продуктом);
- ✓ сбор данных и защитные механизмы (отказо-устойчивость, понятны ли выводимые ошибки, соответствует ли требованиям безопасности).



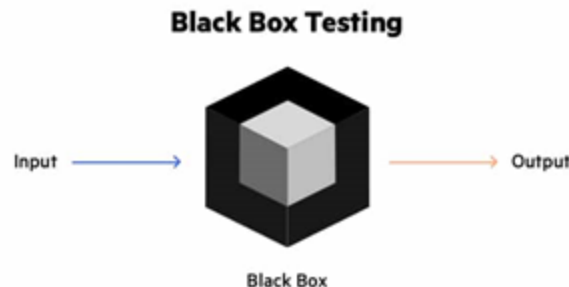


# Тестирование по стратегии чёрного ящика (интуитивное тестирование)

## Цель:

анализ причинно-следственной связи между взаимодействием пользователя с продуктом и результатом, полученным за счёт этого взаимодействия.

Пользователям говорят, для чего предназначен продукт, но изучать, как именно он работает, они смогут самостоятельно.







# Другие классификации тестов



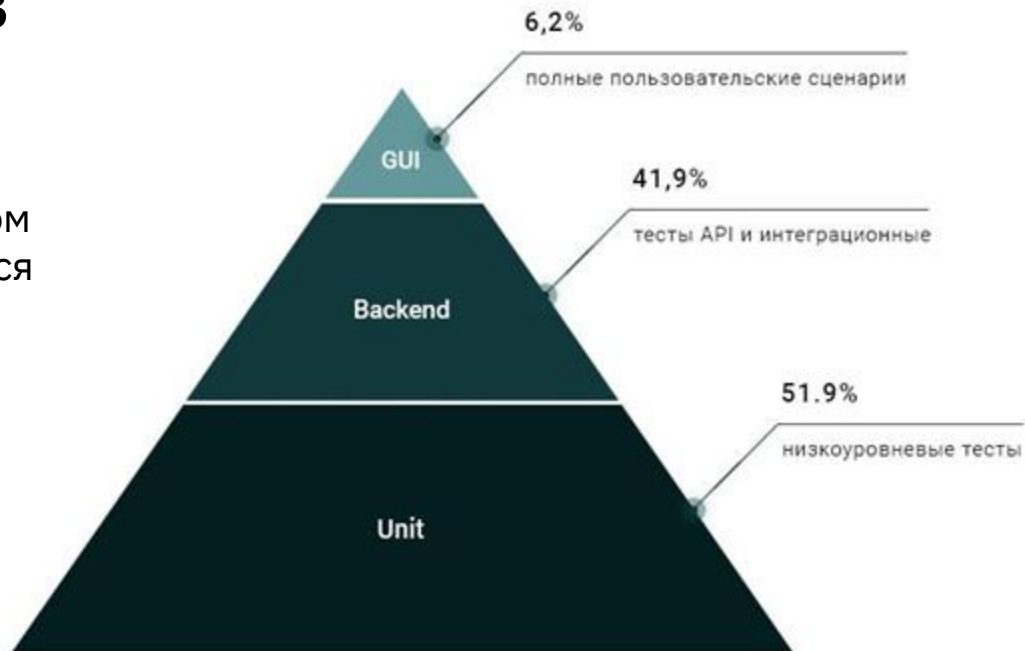
- **функциональные**
  - все типы UAT, которые обсуждали выше
  - интеграционное и низкоуровневое (идёт в течении проекта)
- **нефункциональные**
  - нагрузочное
  - конфигурационное
  - тестирование удобства
- **тестирование связанное с изменениями**
  - регрессионное



# Другие типы тестов

## Интеграционное

тип тестирования, при котором программные модули объединяются логически и тестируются как группа.





# Другие типы тестов

## Нагрузочные

- ✓ эмуляция множества пользователей
- ✓ подвид тестирования производительности, сбор показателей и определение производительности и времени отклика программы или технической системы в ответ на внешний запрос в момент нагруженности системы





Вопрос

Из чего должны состоять тест-планы?





# Формирование тест-планов

**План тестирования** должен содержать в себе:

- ✓ что тестируем, приоритеты
- ✓ условия входа и выхода (когда продукт готов к UAT и когда тестирование будет считаться завершенным)
- ✓ кто будет участвовать в тестировании
- ✓ какая роль будет отводиться участникам тестирования в течение всего процесса
- ✓ график и продолжительность тестирования
- ✓ как будут собираться, анализироваться и задействоваться тестовые данные



The UAT Test Plan





Вопрос

Когда формировать тест-планы?





# Когда формировать тест-планы

- Лучше всего одновременно с подготовкой ТЗ.
- Это помогает при реализации проекта.
- Имея готовый тест-план и понимая, каким образом будет тестироваться продукт проекта, РМ может принимать решения при реализации проекта и чётко видеть, куда двигаться, какие изменения в содержание проекта можно вносить и на что они могут повлиять при итоговой приёмке продукта проекта.





Вопрос

CI / CD – непрерывная интеграция и поставка

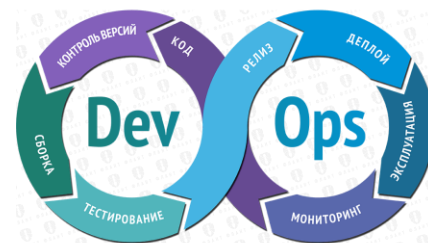






# Непрерывная интеграция (Continuous Integration, CI) Непрерывная поставка (Continuous Delivery, CD)

- это одна из DevOps-практик
- относится и к AGILE-практикам
- автоматизация развёртывания позволяет разработчикам сосредоточиться на реализации бизнес-требований, на качестве кода и безопасности





# Непрерывная интеграция (Continuous Integration, CI)

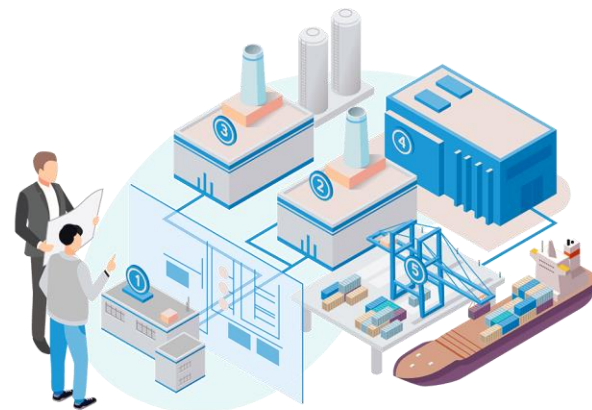
- В код вносятся небольшие изменения, но часто
- Механизм интеграции
- Механизм тестирования вносимых изменений
- Обеспечивает последовательный и автоматизированный способ сборки, упаковки и тестирования





# Непрерывная поставка (Continuous Delivery, CD)

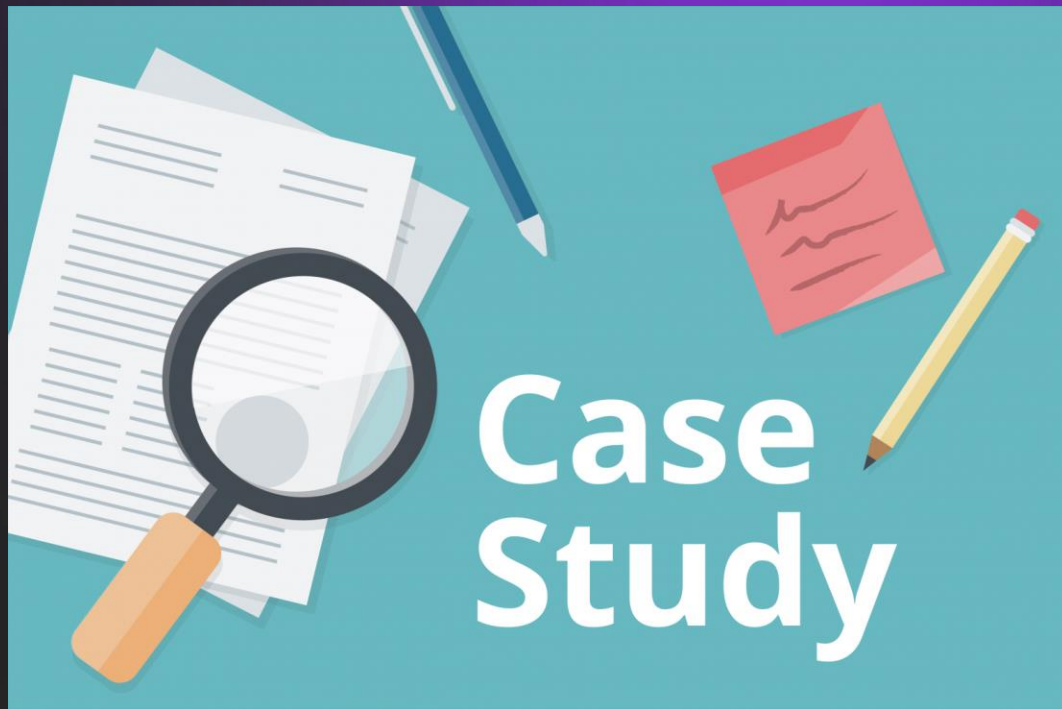
- Стартует после завершения Непрерывной интеграции
- Автоматизирует развёртывание приложения в различных средах







# Кейсы

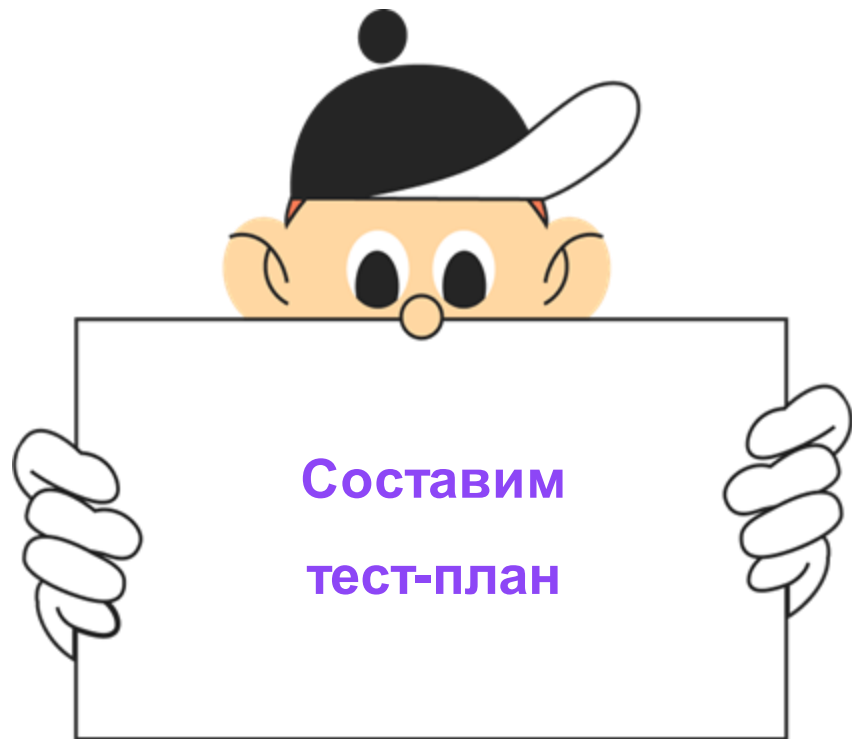




## Кейс

### Проект:

Разработка веб-решения для  
отправки налоговой отчётности в  
электронном виде





Вопросы?

Вопросы?



Вопросы?





## Поделитесь с нами обратной связью

В формате **ДТП**

- Достижения
- Трудности
- Предложения

Описание и материалы

Обсуждение • 0

Задание • 0

Комментарии • 0



Ваш комментарий



# Благодарю за внимание!

Мои контакты:

