

Тестирование программного обеспечения

- Введение в автоматизацию тестирования

(АТ) ПО -

Василий Корпусов
СПБ 2020 г.



Василий Корпусов

- Test Automation Engineer, Bercut ltd
- Автоматизировал 700+ ручных тест-кейсов
- Аспирант кафедры «Автоматизация предприятий связи», СПбГУТ им. Бонч-Бруевича



План

- История развития тестирования
- Процесс тестирования (тестировщик, уровни, виды и типы тестирования)
- Автоматизированное тестирование (автоматизатор)
- Подходы к АТ
- Области применения АТ
- Преимущества и недостатки АТ
- Навыки и умения



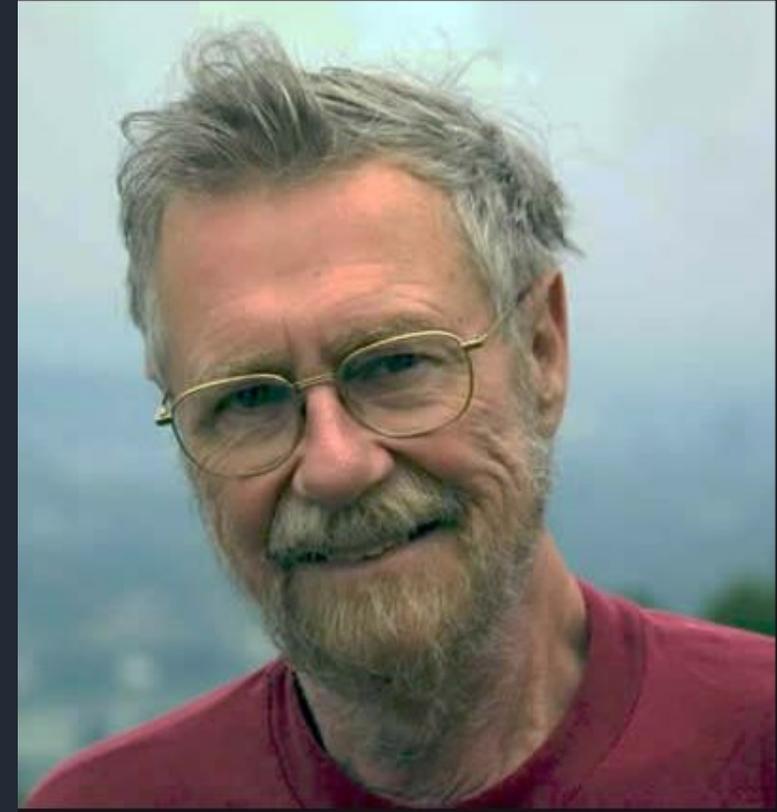
Источники

- ISTQB (International Software Testing Qualifications Board) - <https://www.rstqb.org/ru/istqb-downloads.html>
- ГОСТ Р 56922-2016/ISO/IEC/IEEE 29119-1:2013 Системная и программная инженерия. Тестирование программного обеспечения. Часть 1-3 - <http://docs.cntd.ru/document/1200134996>,



Введение

„Тестирование программы может весьма эффективно продемонстрировать наличие ошибок, но оно не демонстрирует их отсутствие.“



- *Эдсгер Вибе Дейкстра, Ученый-информатик, сооснователь структурного программирования*

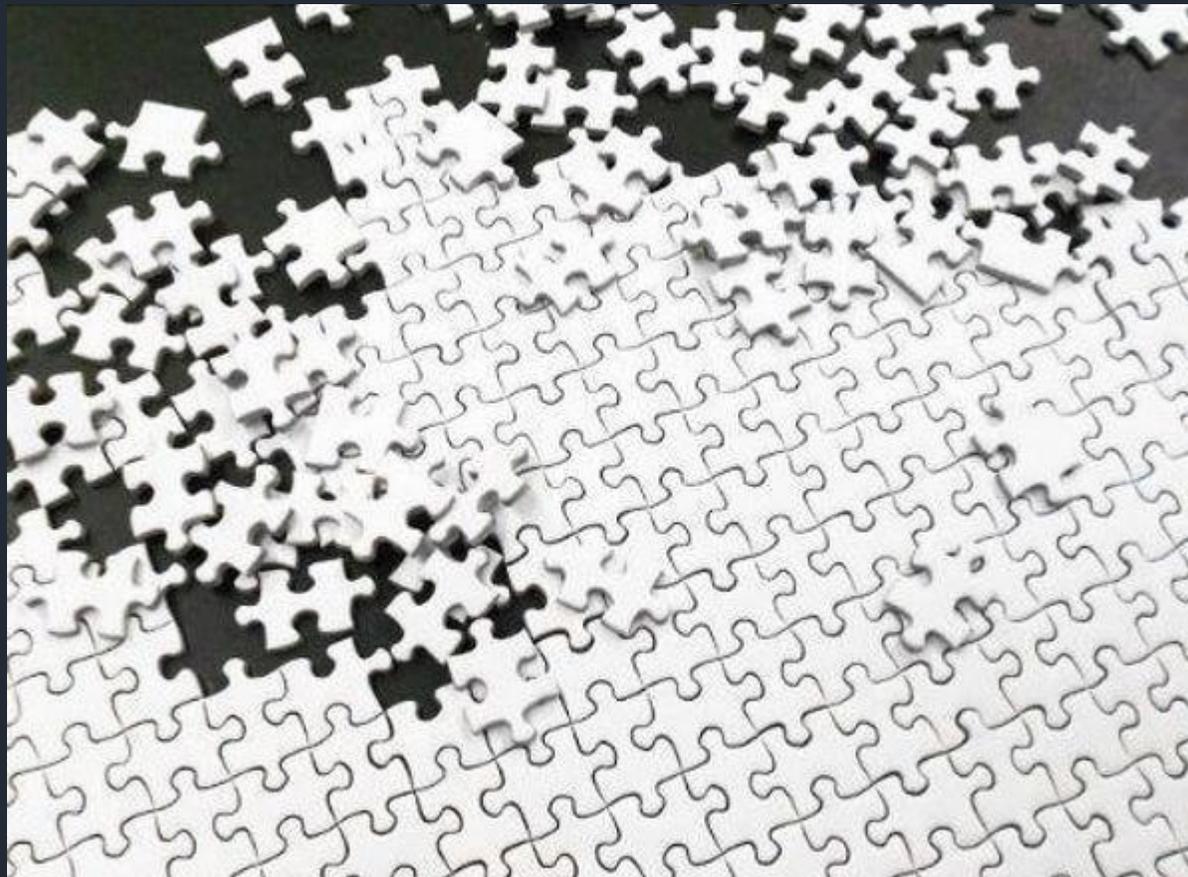


История: Появление процесса тестирования ПО

- ✓ Первые программы разрабатывались для МО
- ✓ Строго формализованное тестирование
- ✓ Отдельный процесс тестирования



Тестирование в



- В 1960-х много внимания уделялось «исчерпывающему» тестированию, которое должно проводиться с использованием всех путей в коде или всех возможных входных данных

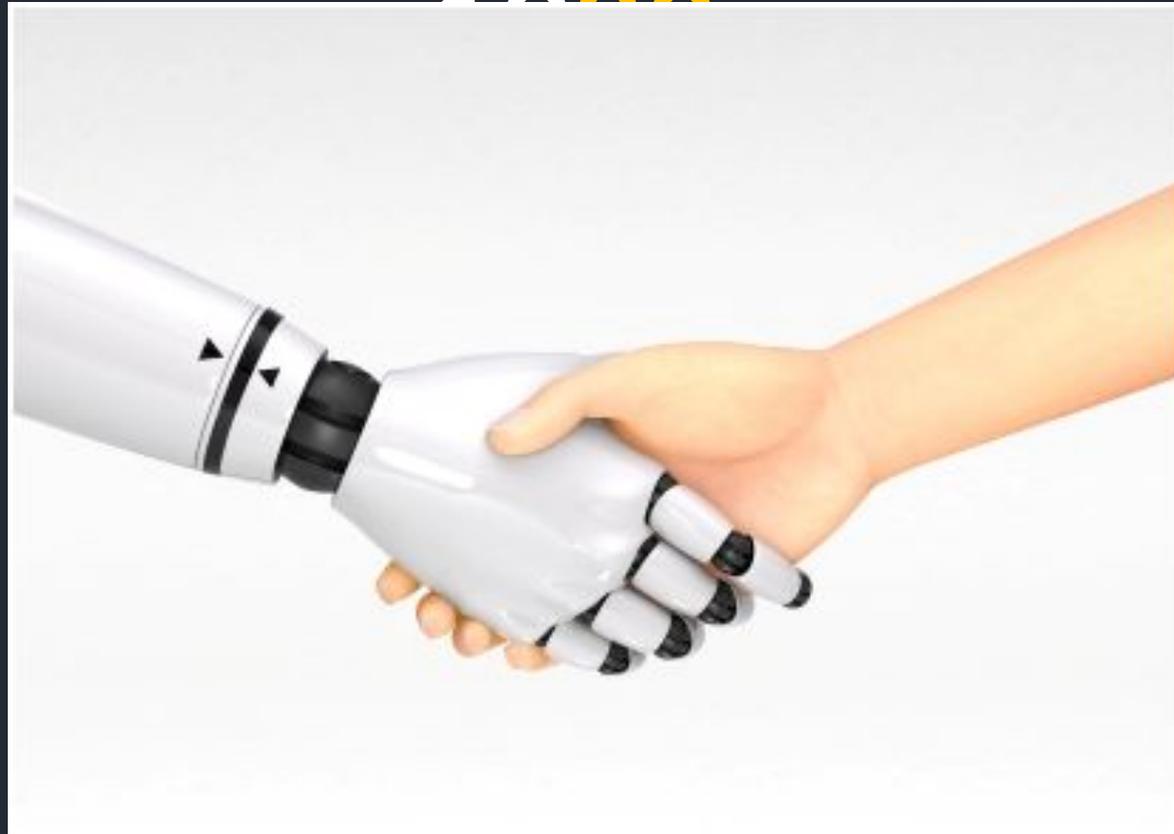
Тестирование в



- В начале 1970-х тестирование ПО обозначалось как «процесс, направленный на демонстрацию корректности продукта»



Тестирование в



- В **1980-х** тестирование расширилось таким понятием, как предупреждение дефектов
- В **середине 1980-х** появились первые инструменты для **автоматизированного** тестирования

Тестирование в 1990-х



- В **начале 1990-х** переход от тестирования к обеспечению качества, охватывающего весь цикл разработки ПО
- В **середине 1990-х** с развитием Интернета и разработкой большого количества веб-приложений особую популярность стало получать «гибкое тестирование»

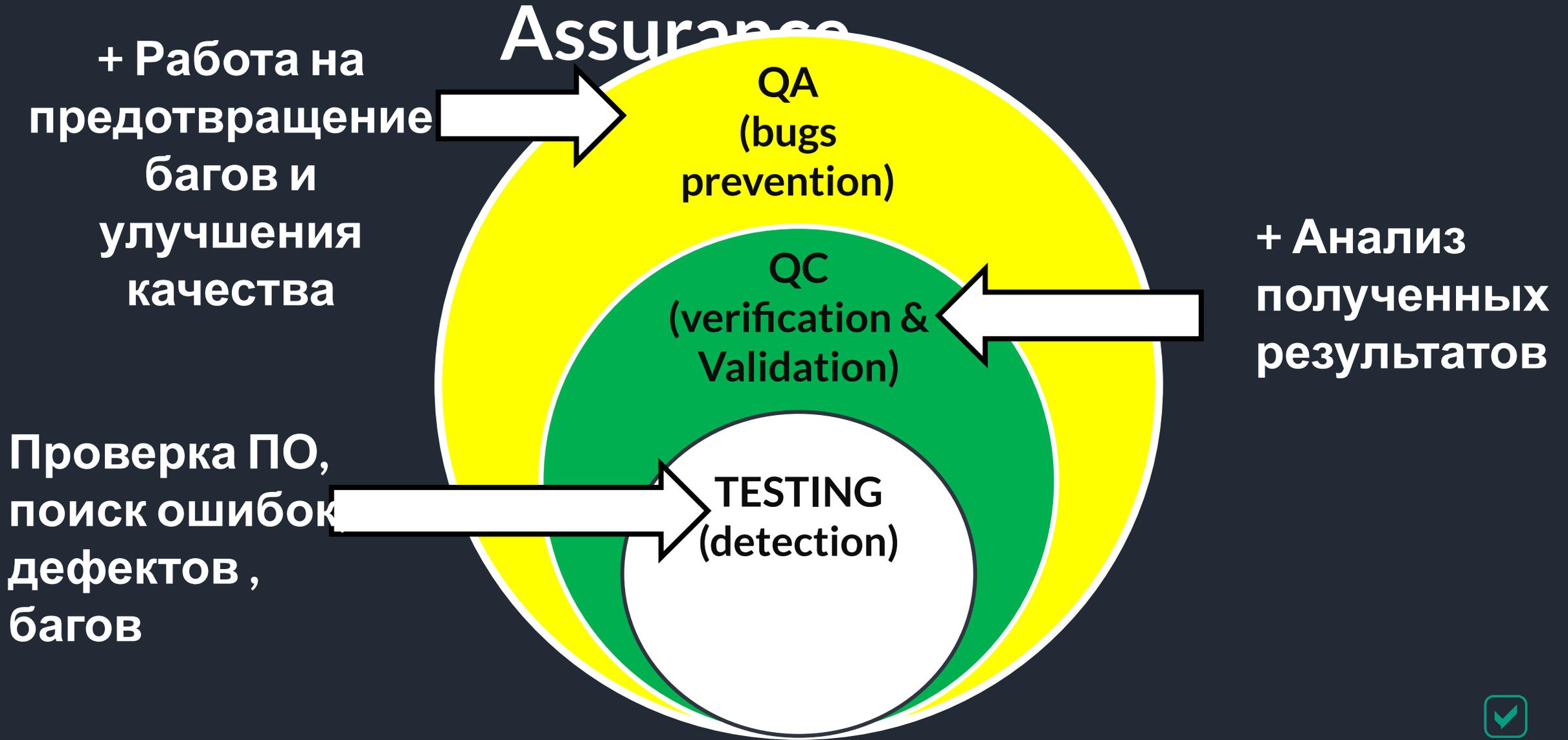


Тестирование в



- В **2000-х** появилось еще более широкое определение тестирования, когда в него было добавлено понятие «оптимизация бизнес-технологий» (en:business technology optimization, BTO). BTO направляет развитие информационных технологий в соответствии с целями бизнеса.

Testing - Quality Control - Quality Assurance



Software Testing

Software Testing (тестирование ПО) — процесс исследования, испытания программного продукта, имеющий своей целью проверку соответствия между реальным поведением программы и её ожидаемым поведением на конечном наборе тестов, выбранных определённым образом ([ISO/IEC TR 19759:2005](#))



Quality Control

Quality Control (контроль качества) — это процесс нахождения ошибок в продукте, с целью их последующего исправления.

- Задачей **Quality Control** является поддержка качества продукта в текущий момент времени.

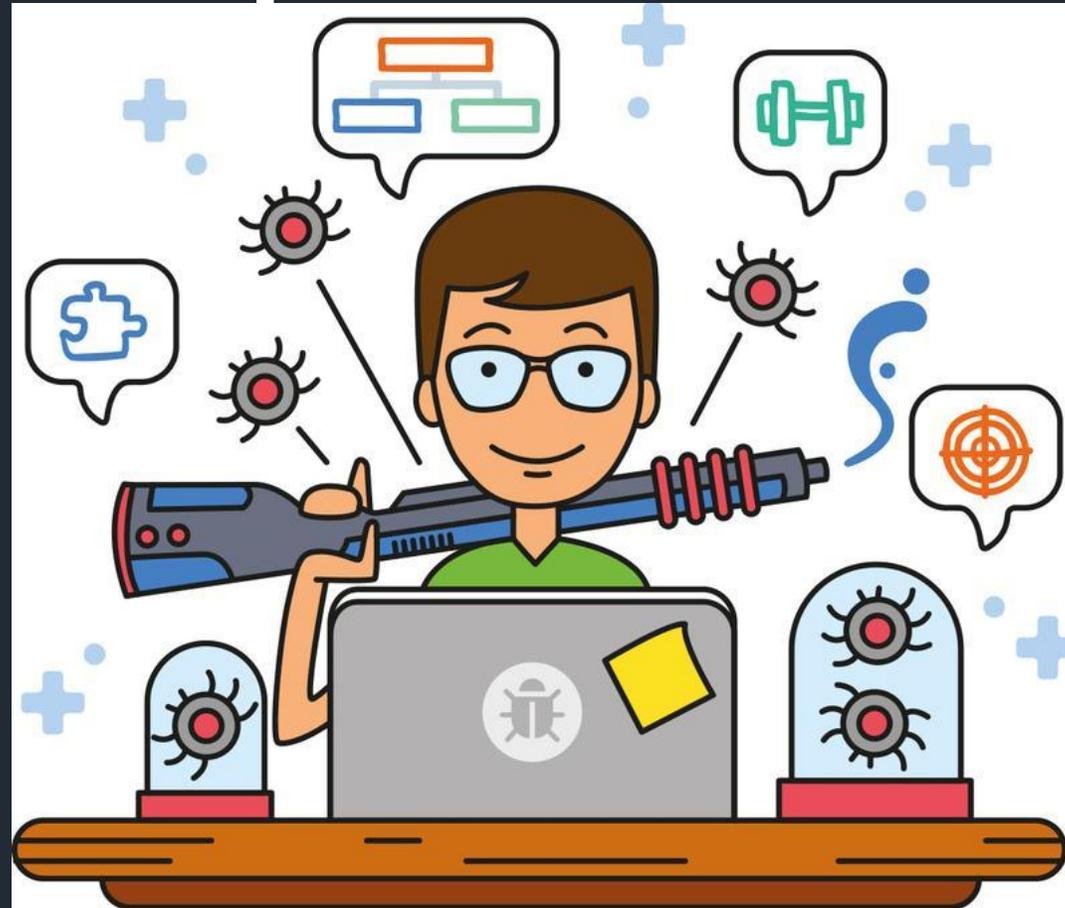


Quality Assurance



Quality Assurance (обеспечение качества) — это превентивный процесс, задачей которого является обеспечение качества продукта в будущем.

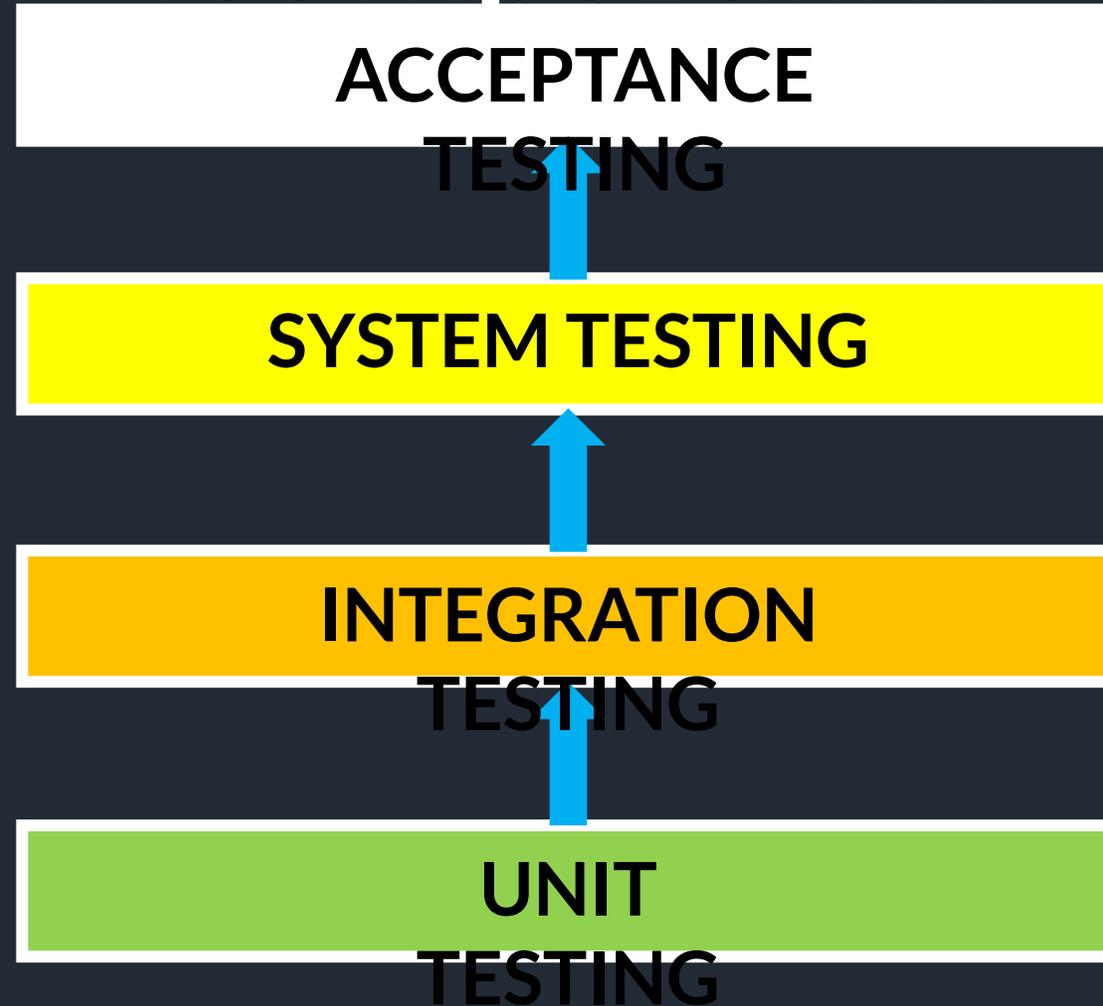
Тестировщик – QA/QC



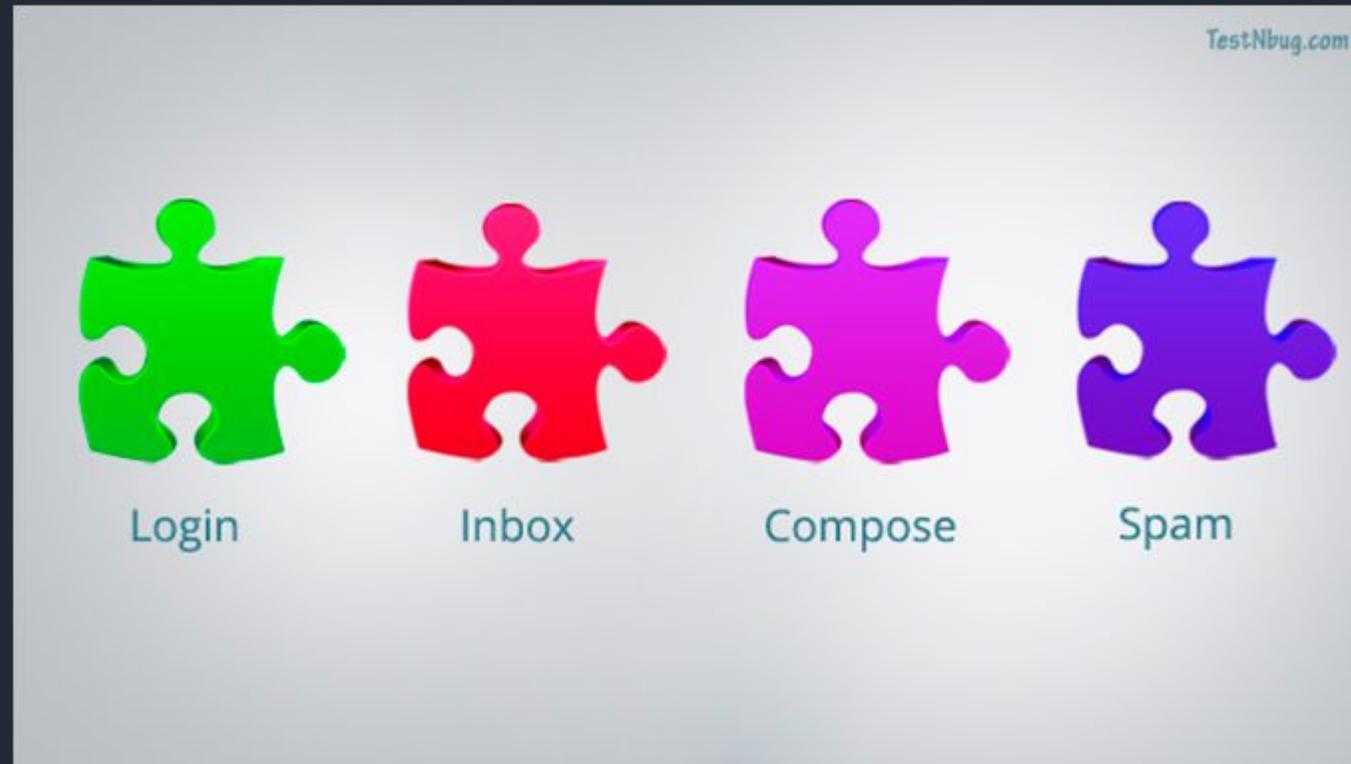
Тестировщик (tester) – это опытный специалист, принимающий участие в тестировании компонента или системы. [ISTQB Glossary]

Уровни

тестирования



Unit Testing

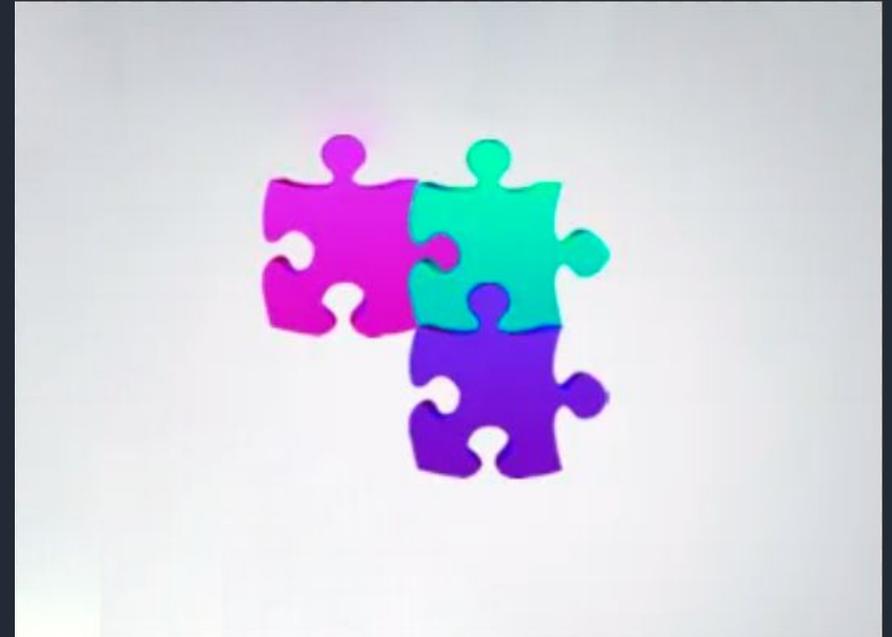


Модульное(компонентное) тестирование - тестирование отдельных компонентов программного обеспечения.

[Согласно IEEE 610]

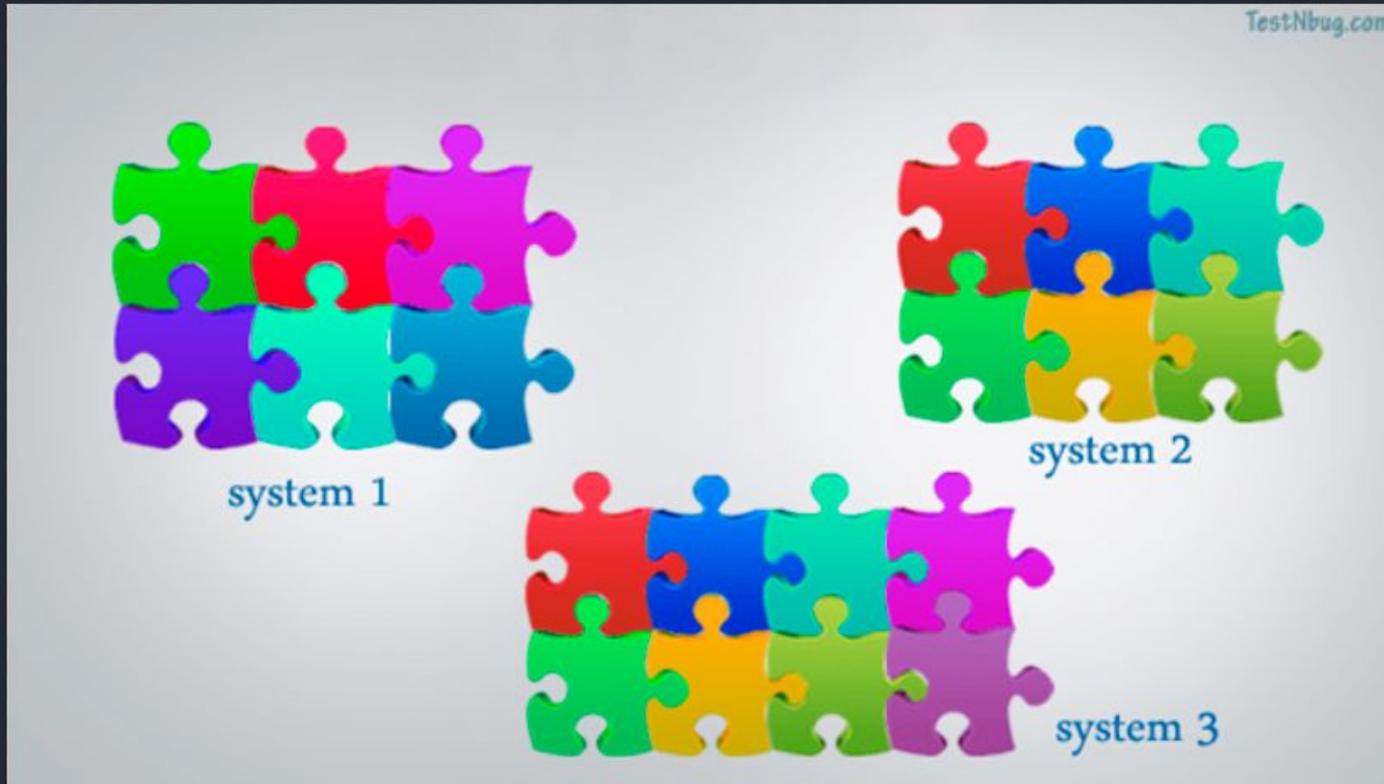
Integration Testing

- **Component Integration Testing** (компонентное тестирование)
- **System Integration Testing** (Системное тестирование)



Интеграционное тестирование - тестирование, выполняемое для обнаружения дефектов в интерфейсах и во взаимодействии между интегрированными компонентами или системами. [\[ISTQB Glossary\]](#)

System Testing



□ Alpha Testing
(альфа
тестирование)

□ Beta Testing
(бета тестирование)

Системное тестирование - процесс тестирования системы в целом с целью проверки того, что она соответствует установленным требованиям. [\[ISTQB Glossary\]](#)

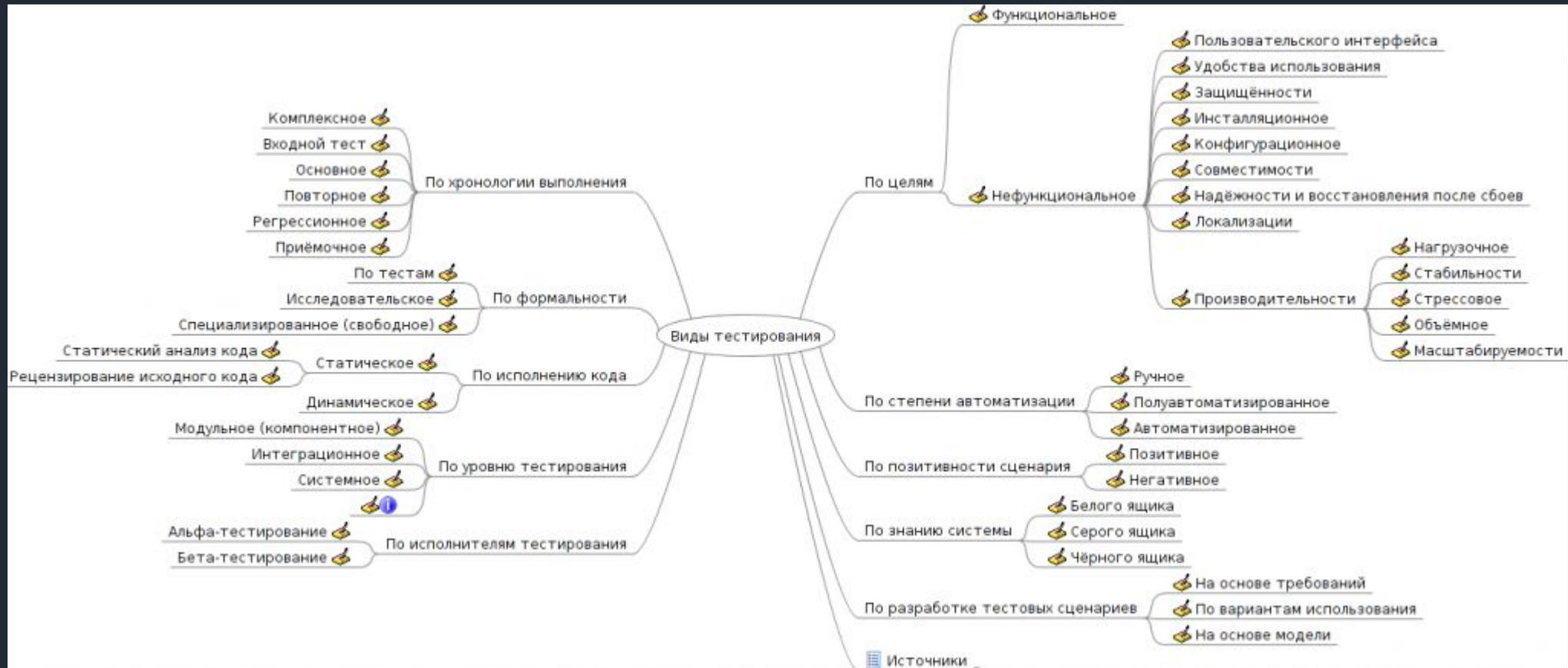


Acceptance Testing



Приёмочное тестирование - проводится с целью определения соответствия системы критериям приёмки и дать возможность пользователям, заказчикам или иным авторизованным лицам определить, принимать систему или нет. [Согласно IEEE 610]

Виды тестирования



По объекту

ТЕСТИРОВАНИЯ

- **Functional testing**
- GUI testing
- Localization testing
- Performance testing
- Security testing
- Usability testing
- Compatibility testing



Functional Testing



Проводится в двух аспектах:

- Требования
- Бизнес-процессы

Функциональное тестирование - тестирование, основанное на анализе спецификации функциональности компонента или системы. [ISTQB Glossary]

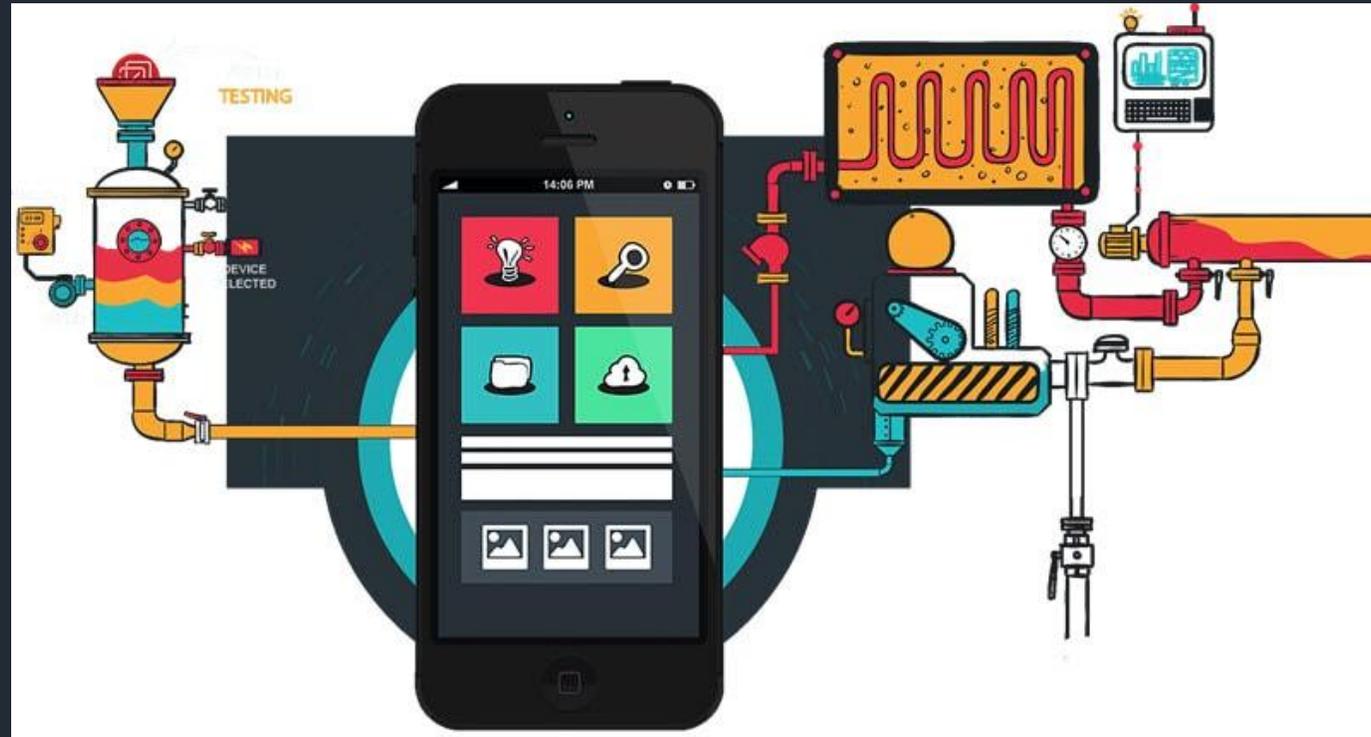
По объекту

ТЕСТИРОВАНИЯ

- Functional testing
- GUI testing**
- Localization testing
- Performance testing
- Security testing
- Usability testing
- Compatibility testing



GUI Testing



Тестирование пользовательского интерфейса – функциональная проверка интерфейса на соответствие требованиям (размер, шрифт, цвет и т.д.).

По объекту

ТЕСТИРОВАНИЯ

- Functional testing
- GUI testing
- Localization testing**
- Performance testing
- Security testing
- Usability testing
- Compatibility testing



Localization Testing

Тестирование локализации - это процесс тестирования локализованной версии программного продукта.
[ISTQB Glossary]



По объекту

ТЕСТИРОВАНИЯ

- Functional testing
- GUI testing
- Localization testing
- Performance testing**
- Security testing
- Usability testing
- Compatibility testing



Performance Testing



По объекту

ТЕСТИРОВАНИЯ

- Functional testing
- GUI testing
- Localization testing
- Performance testing
- Security testing**
- Usability testing
- Compatibility testing



Security Testing



Тестирование безопасности - тестирование с целью оценить защищенность программного продукта. [\[ISTQB Glossary\]](#)



По объекту

ТЕСТИРОВАНИЯ

- Functional testing
- GUI testing
- Localization testing
- Performance testing
- Security testing
- Usability testing**
- Compatibility testing



Usability Testing



Тестирование удобства пользования - тестирование с целью определения степени понятности, легкости в изучении и использовании, привлекательности программного продукта для пользователя. [ISO 9126]

По объекту

ТЕСТИРОВАНИЯ

- Functional testing
- GUI testing
- Localization testing
- Performance testing
- Security testing
- Usability testing
- Compatibility testing**



Compatibility Testing

- Обратная совместимость
- Прямая совместимость



Тестирование совместимости - процесс тестирования для определения возможности взаимодействия программного продукта. [\[ISTQB Glossary\]](#)

По запуску кода на исполнение

□ Static testing

(статическое тестирование)

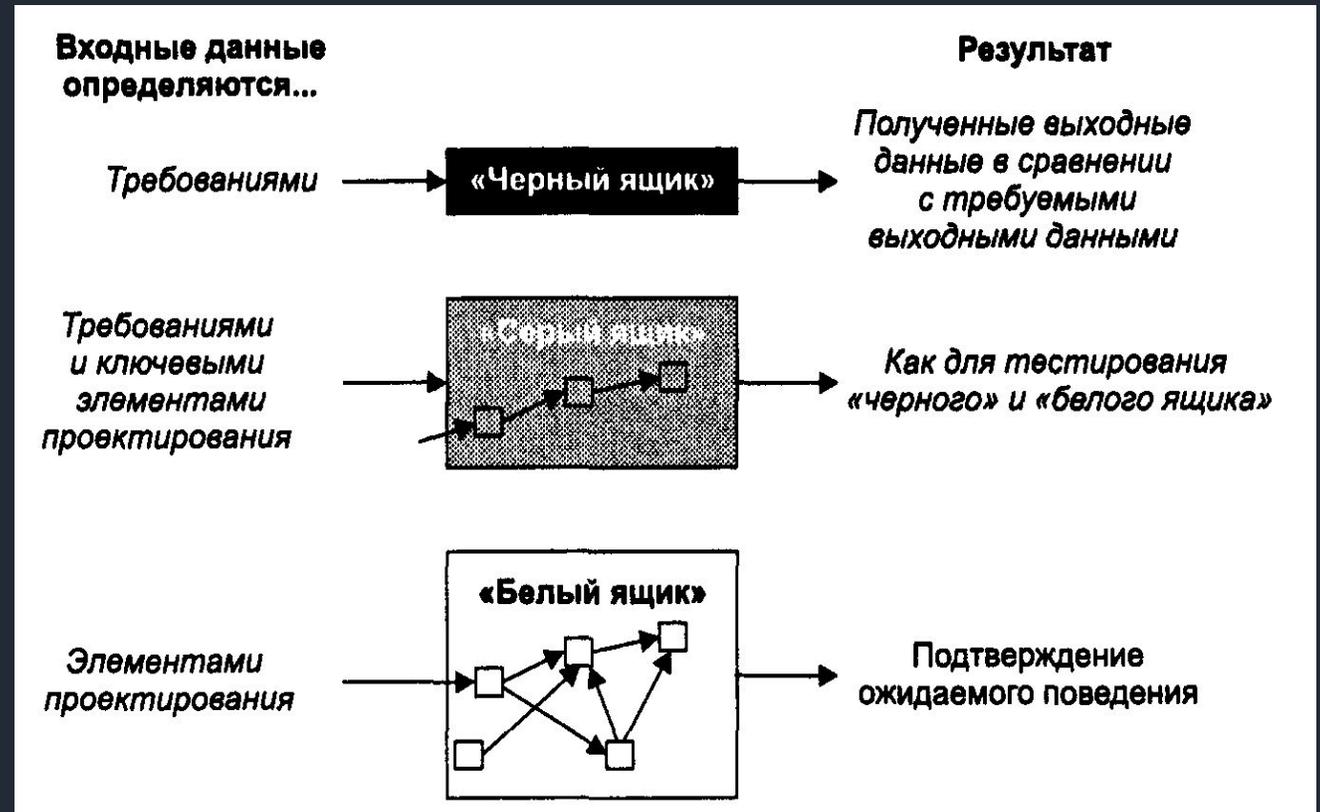
□ Dynamic testing

(динамическое тестирование)



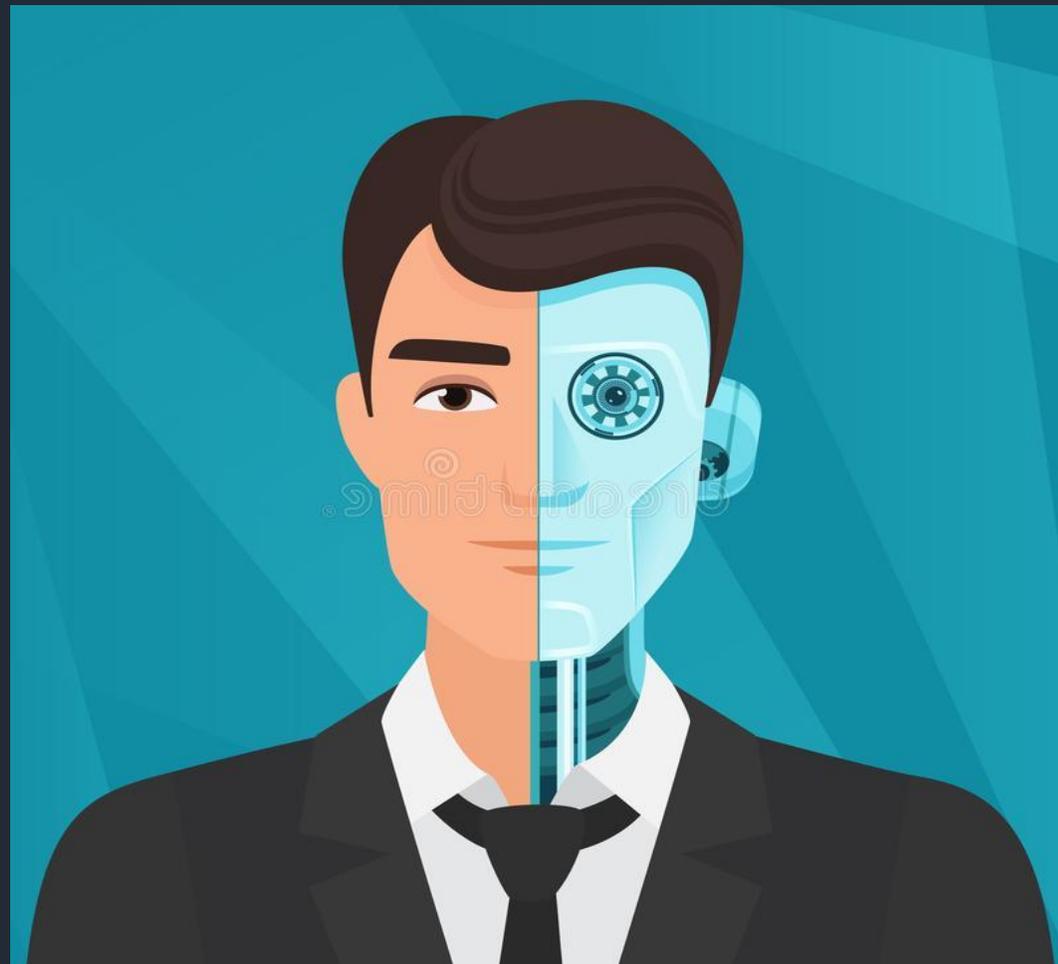
По доступу к коду

- **Black box**
(system test)
- **Grey box**
(integration testing)
- **White box**
(unit testing)



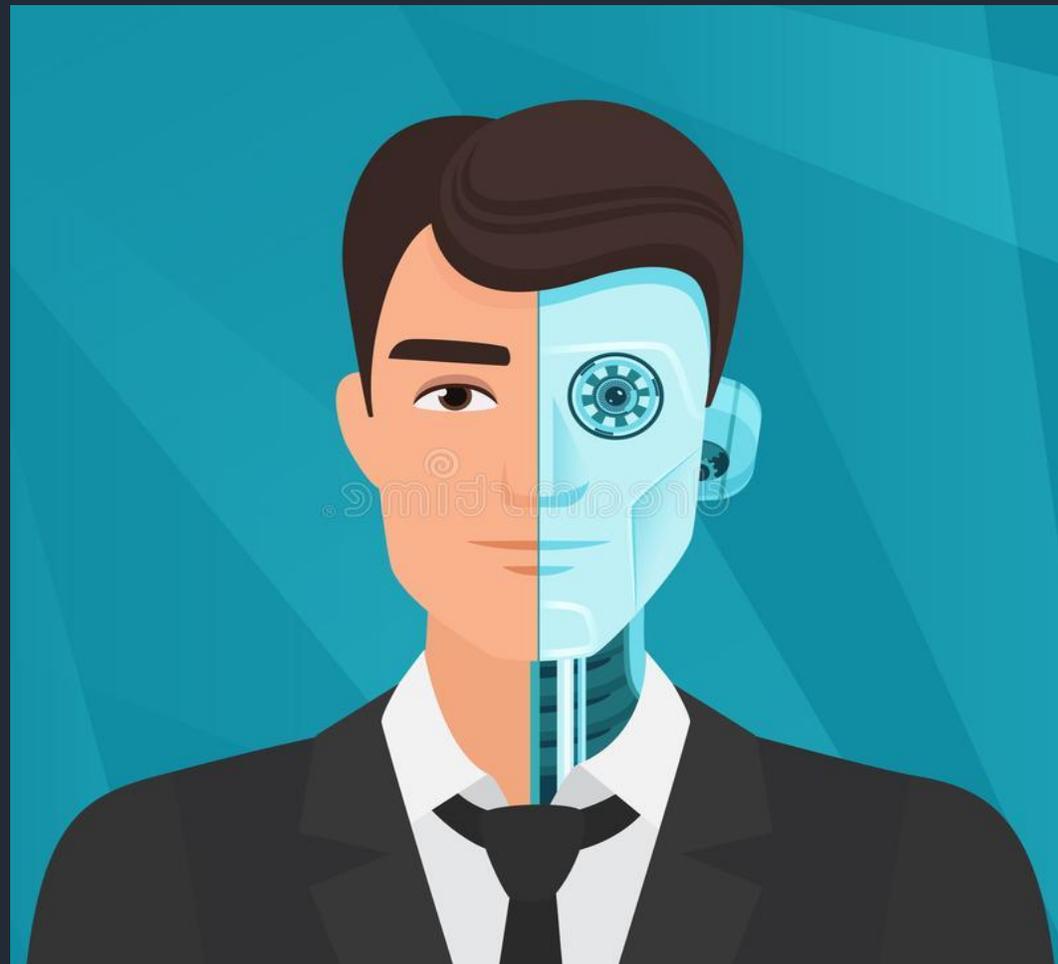
По степени автоматизации

- Manual testing**
- Automated testing
- Semi automated testing



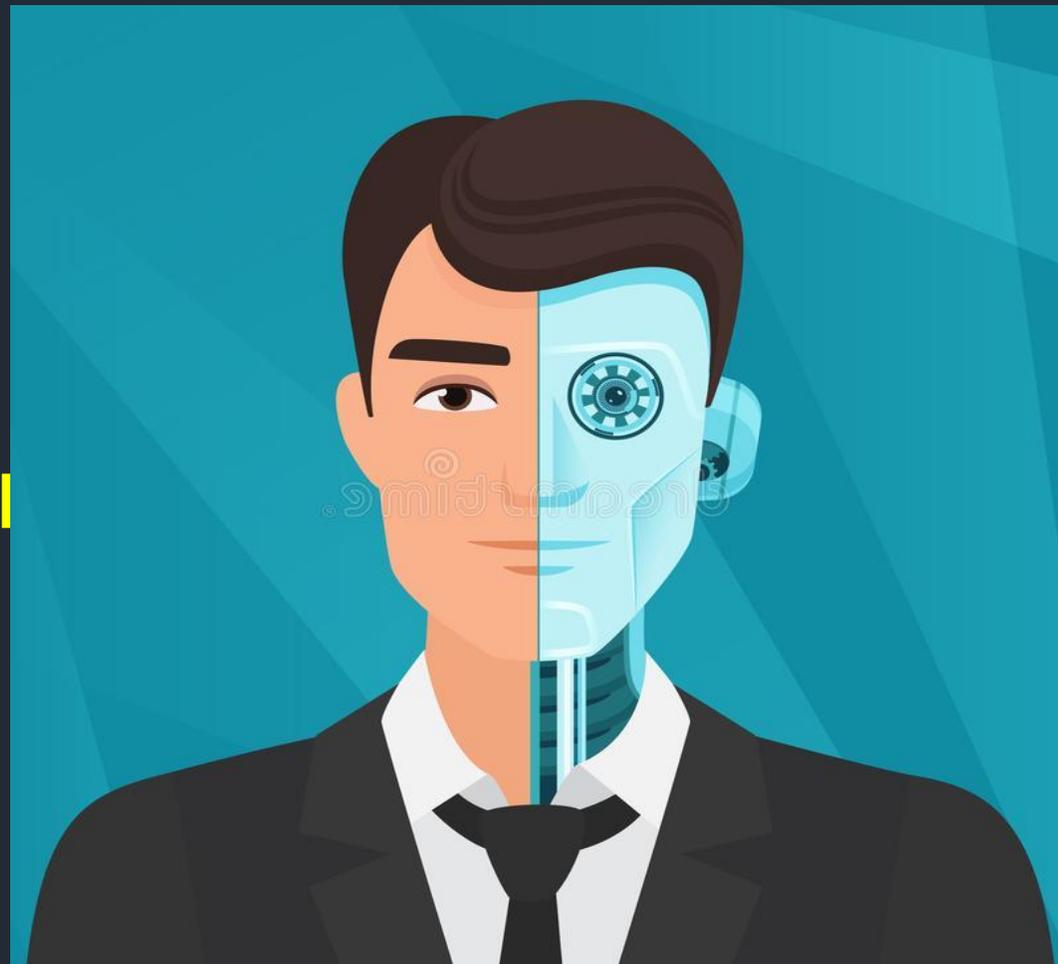
По степени автоматизации

- Manual testing
- Automated testing**
- Semi automated testing



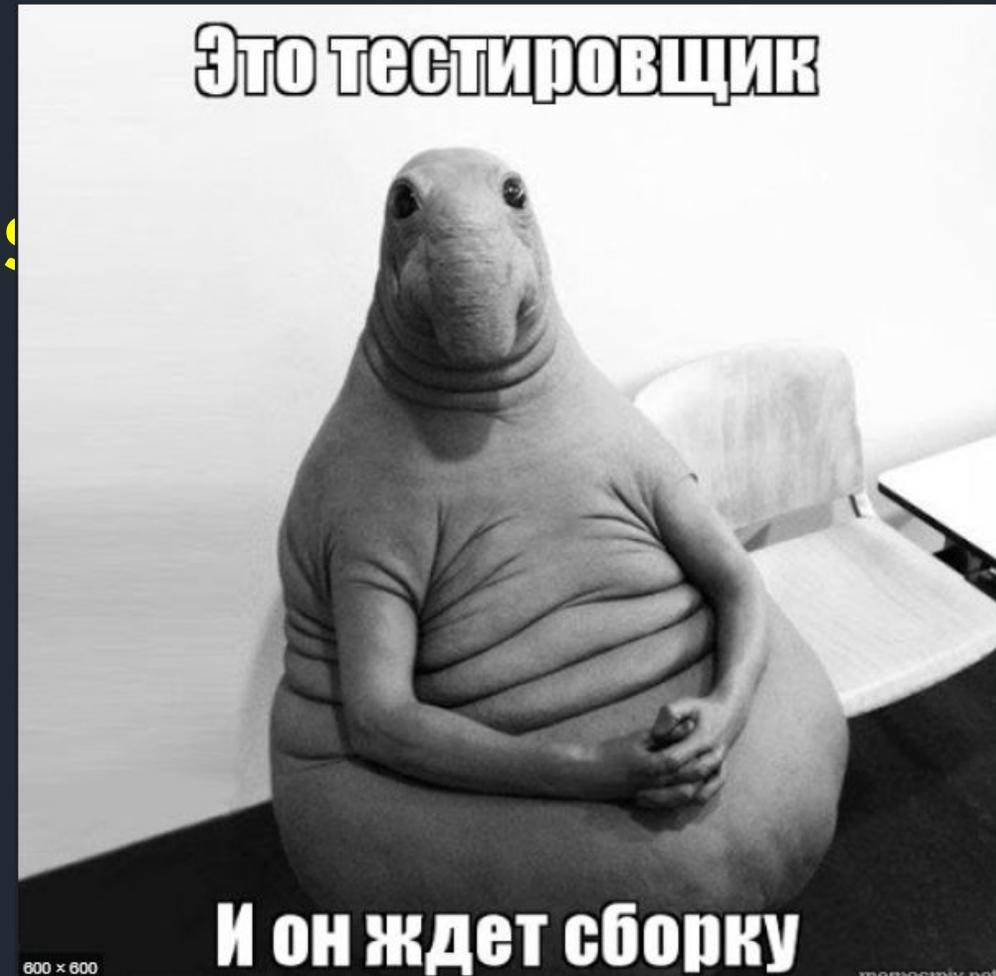
По степени автоматизации

- Manual testing
- Automated testing
- Semi automated testing



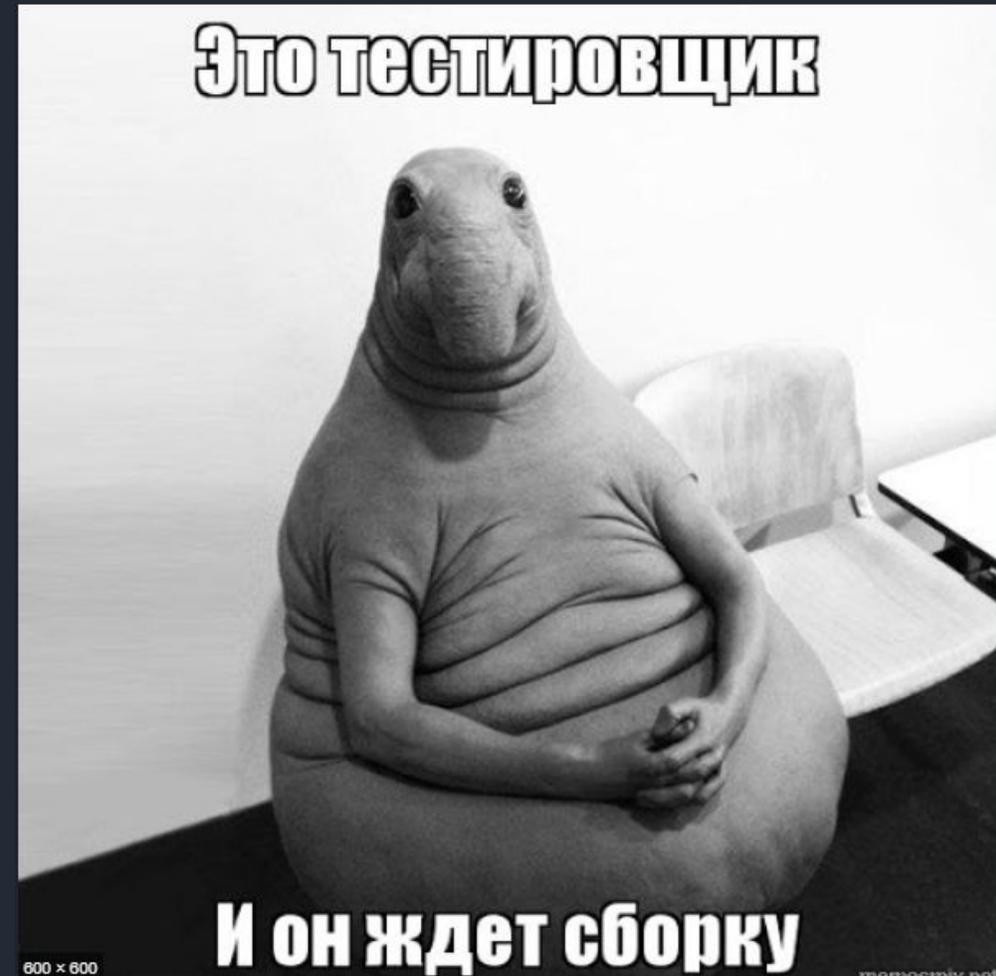
Связанное с изменениями

- **Build Verification Test**
- Smoke Testing
- Sanity Testing
- Regression Testing



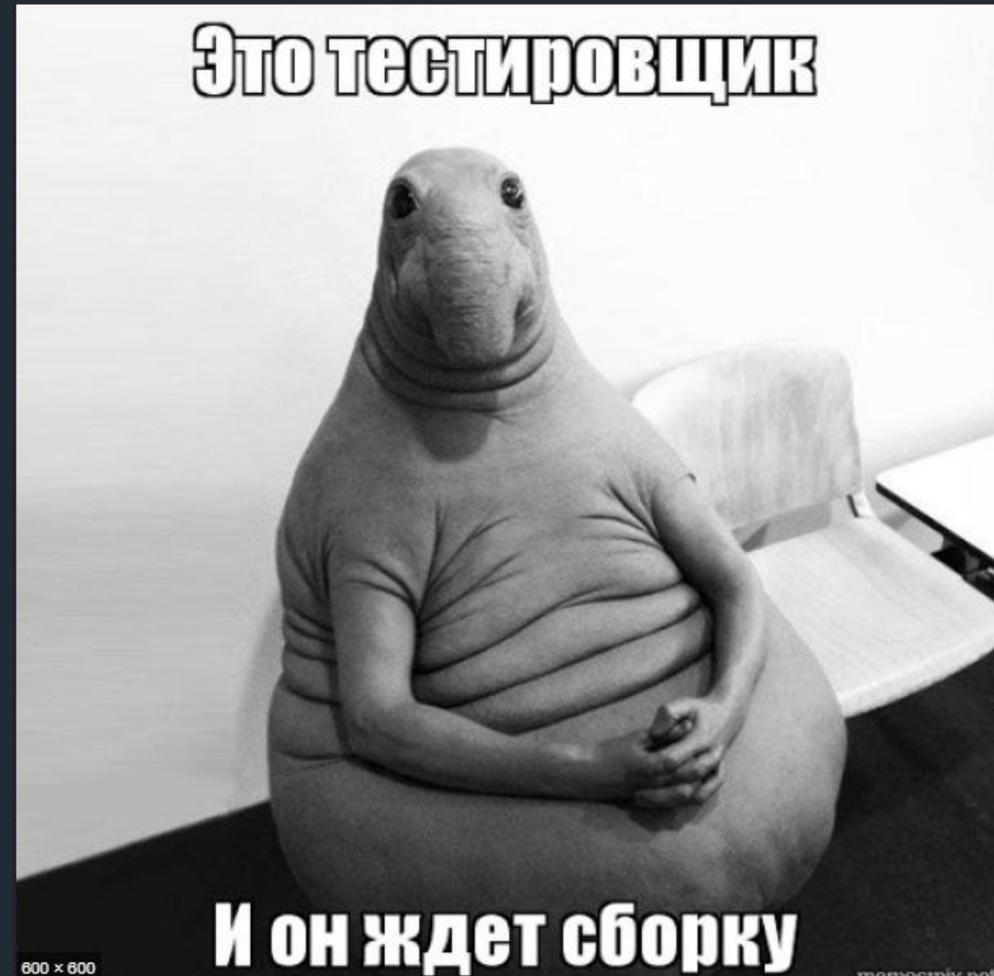
Связанное с изменениями

- Build Verification Test
- Smoke Testing**
- Sanity Testing
- Regression Testing



Связанное с изменениями

- Build Verification Test
- Smoke Testing
- Sanity Testing**
- Regression Testing



Связанное с изменениями

- Build Verification Test
- Smoke Testing
- Sanity Testing
- Regression Testing



По признаку позитивности сценар

- ☐ Positive testing
- ☐ Negative testing



По признаку позитивности сценар

- Positive testing
- Negative testing



Тестирование по плану



Тестирование без



- **Ad hoc testing** (свободное тестирование)
- **Exploratory testing** (исследовательское тестирование)

Типы тестирования

- Smoke test**
- Critical path test
- Extended test



Smoke test

Пример Smoke-теста для калькулятора:

1. Ввести число
2. Выбрать функцию(сложение, вычитание, умножение и т.д.)
3. Ввести второе число
4. Нажать на равно
5. Убедиться в наличии ответа



Типы тестирования

- Smoke test
- Critical path test**
- Extended test



Critical path test

Примеры шагов критического теста для тестирования калькулятора:

- ✓ Проверка сложения, умножения, деления и т.д.
- ✓ Деление на ноль
- ✓ Оставить пустым
- ✓ Специальные символы
- ✓ Ноль вначале, без разделяющего символа
- ✓ Пробелы
- ✓ Ноль после разделяющего символа для десятичных значений
- ✓ И т.д.



Типы тестирования

- Smoke test
- Critical path test
- Extended test**



Extended test

Примеры шагов расширенного теста для тестирования калькулятора:

- ✓ Нажатие нескольких кнопок подряд (сначала сложение, потом вычитание)
- ✓ Вычисление результата уравнения
- ✓ И т.д.



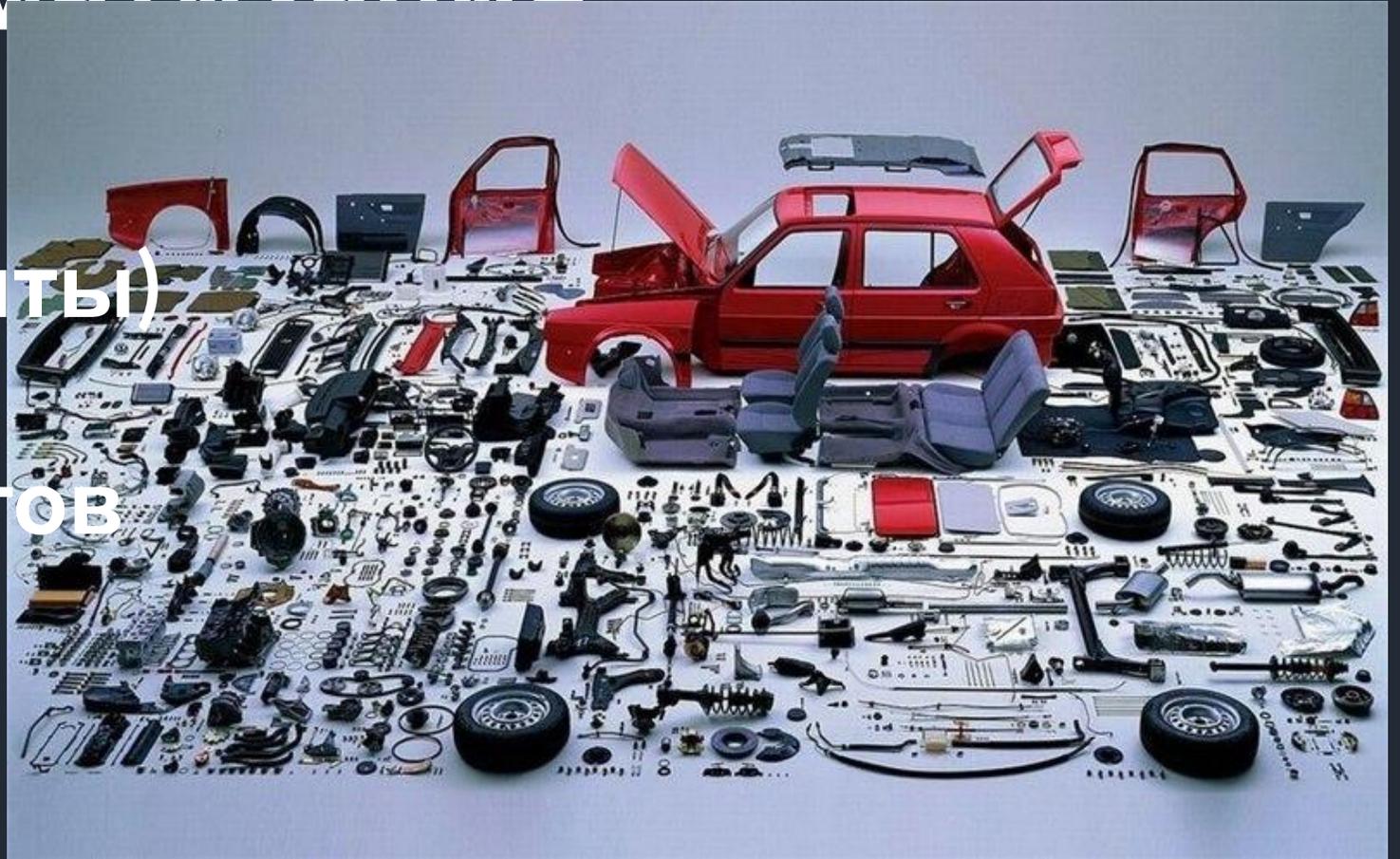
Test Automation



- ❑ **Автоматизация тестирования (test automation):** Использование программного обеспечения для осуществления или помощи в проведении определенных тестовых процессов, например, управление тестированием, проектирование тестов, выполнение тестов и проверка результатов. [[ISTQB Glossary](#)]

Из чего состоит автоматизация?

- Автотесты (скрипты)
- Система запуска
- Анализ результатов

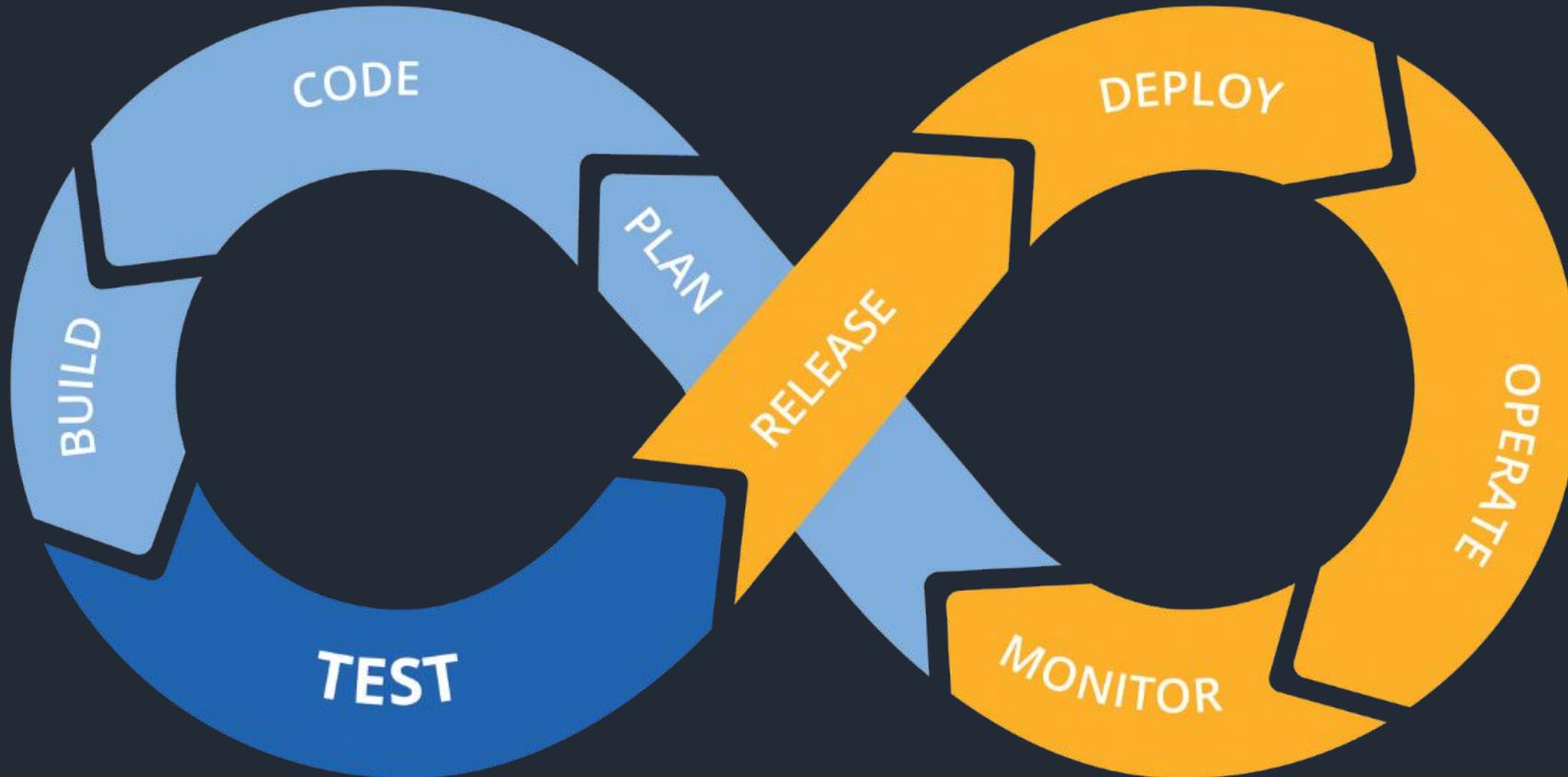


Когда необходима автоматизация?

- ❑ Масштабное приложение с большим количеством бизнес-функций
- ❑ Значительный срок жизни приложения (от 1 года и более)
- ❑ Внедрение CI/CD, регулярные релизы + небольшое количество QA специалистов



Концепция непрерывной интеграции и доставки(CI/CD)



Зачем нужна автоматизация?

**Решение
рутинных
задач**

**Ускорение
выполнения
тестирования**



**Уменьшение
затрат на
ручное
тестирование**

**Увеличение
тестового
покрытия**



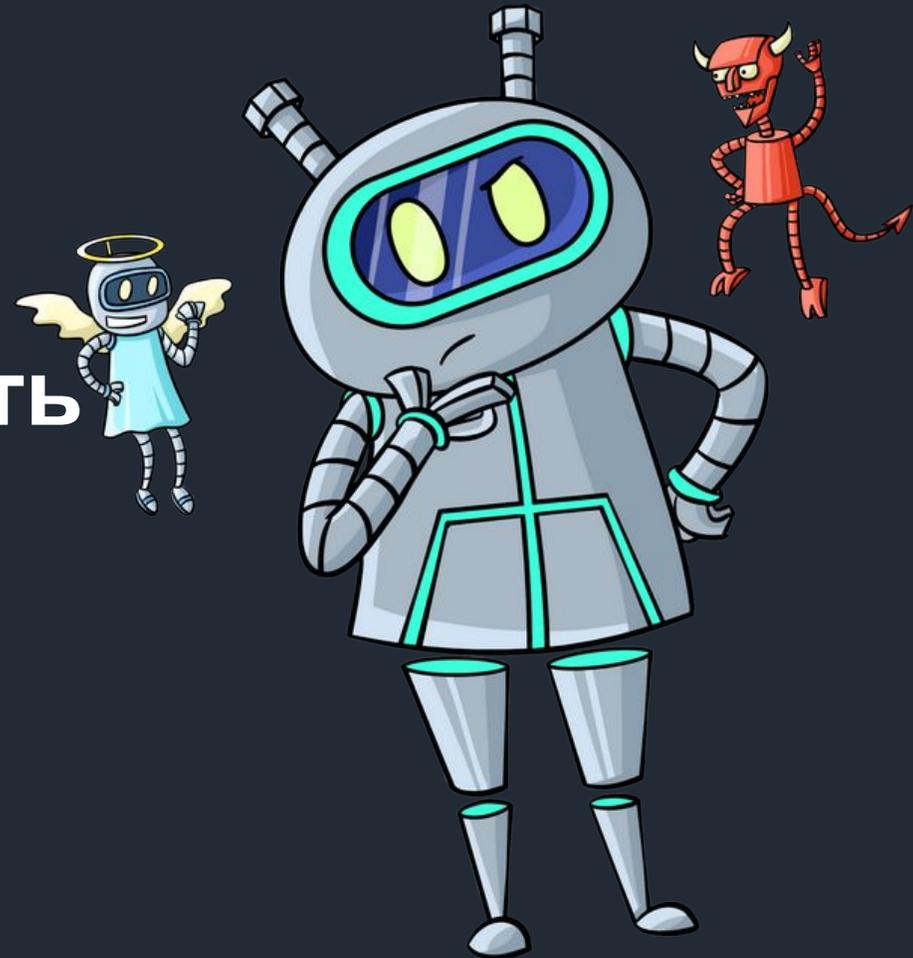
Тестировщик-Автоматизатор



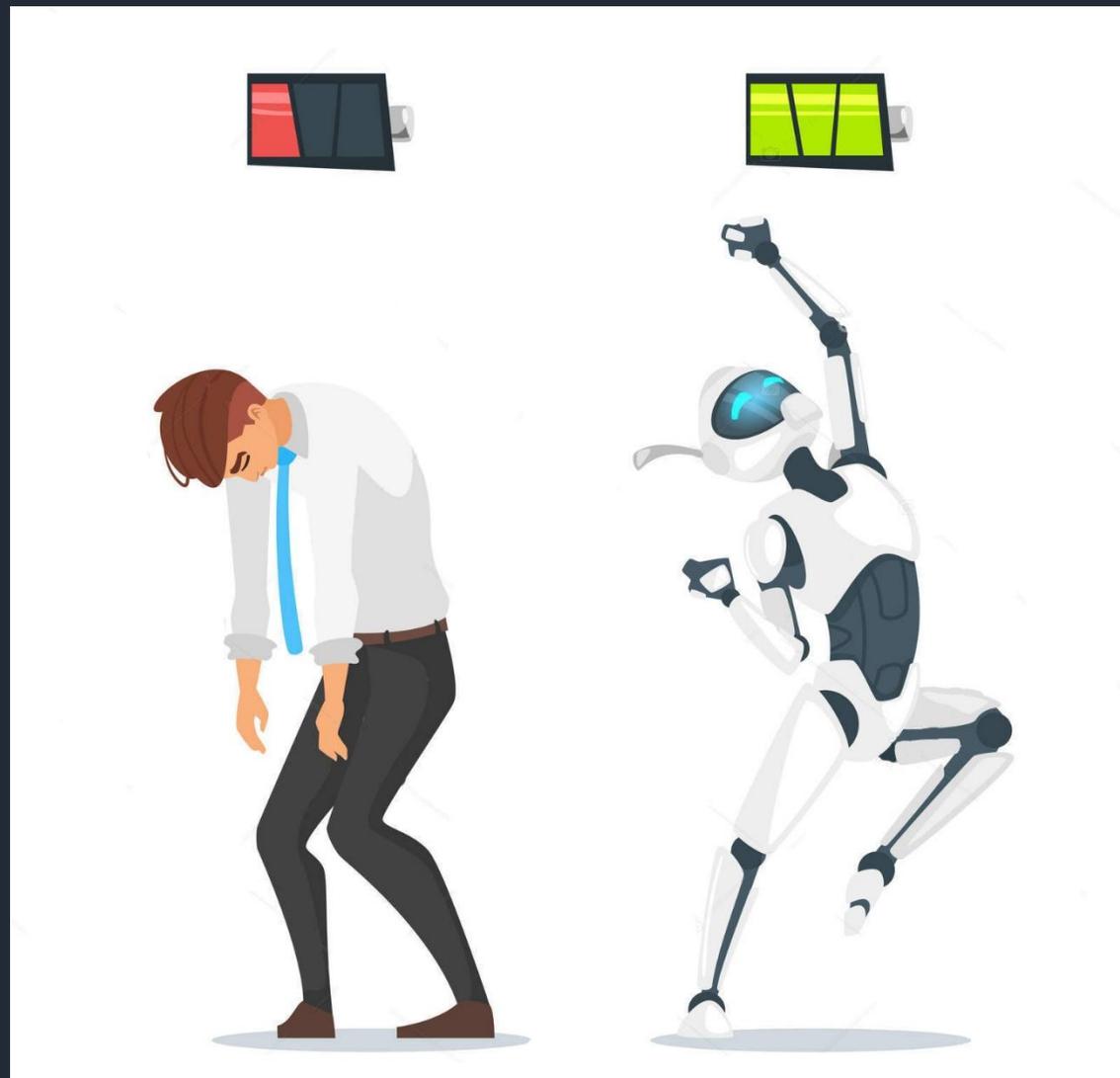
«Типичный Автоматизатор — хитрец. Он сделает все для того, чтобы не делать одно и то же несколько раз».

Плюсы автоматизации

- Неутомимость
- Скорость
- Многофункциональность
- Масштаб
- Экономия сил
- Экономия средств

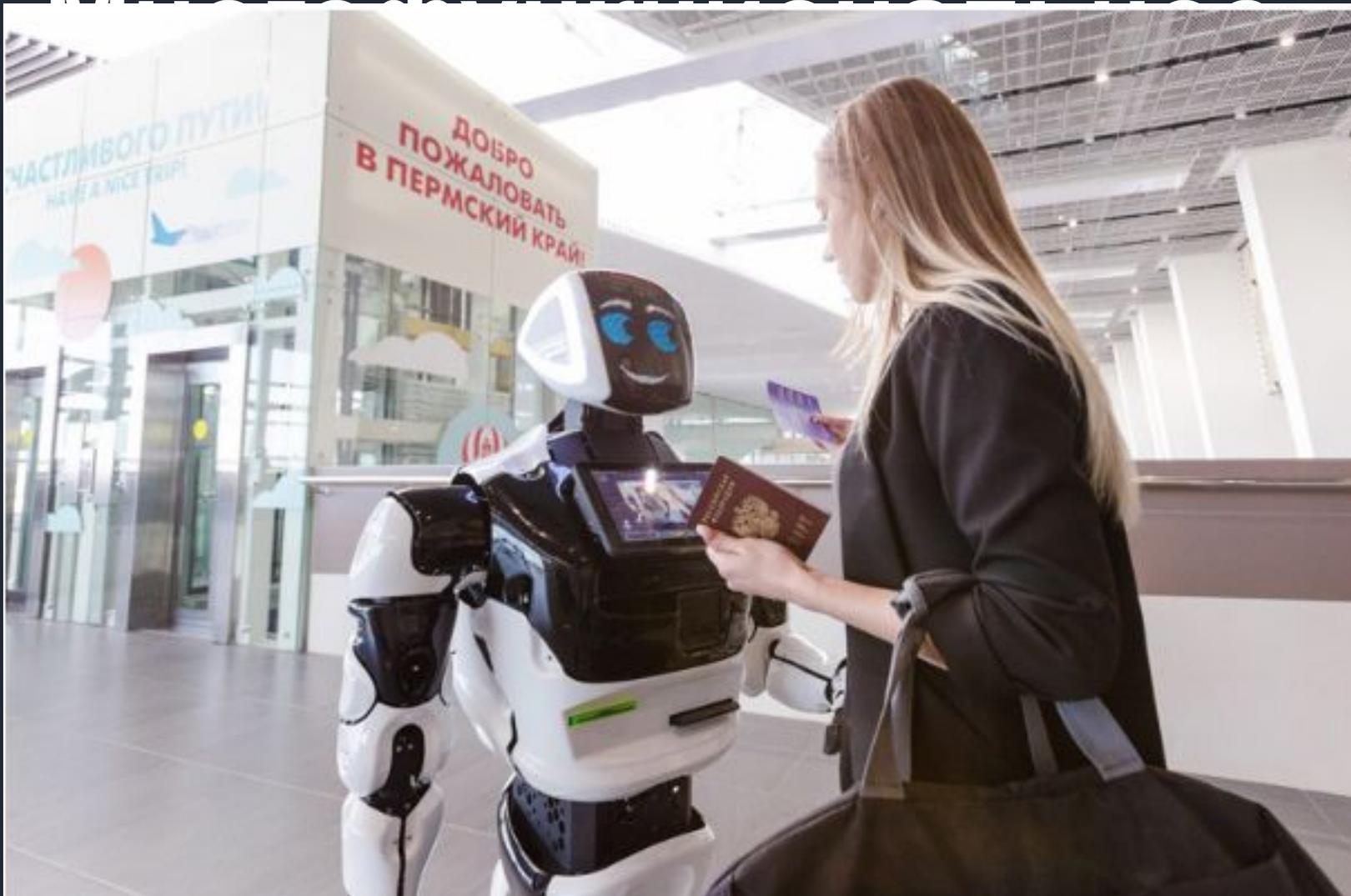


+ Неутомимость

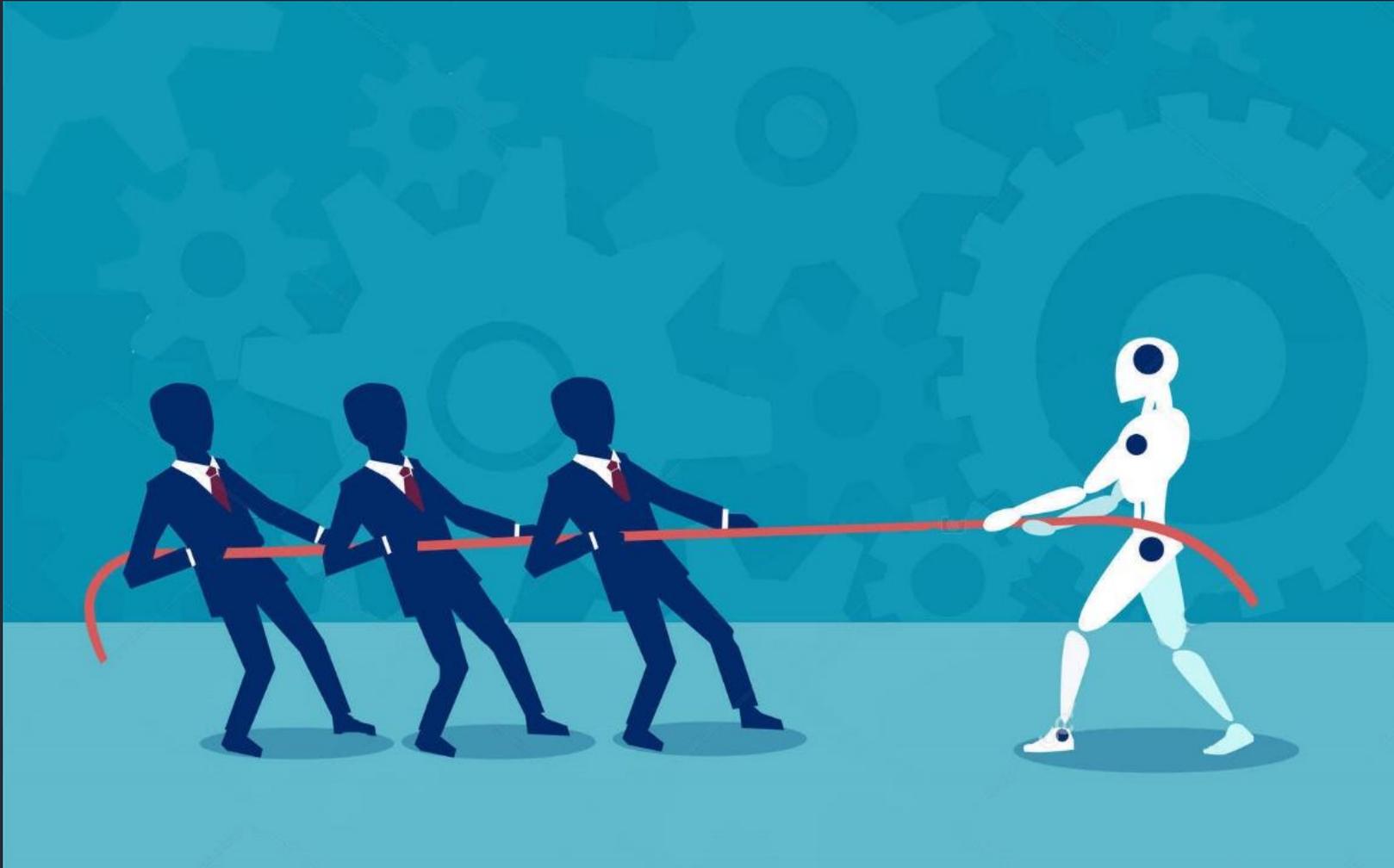


+ Скорость

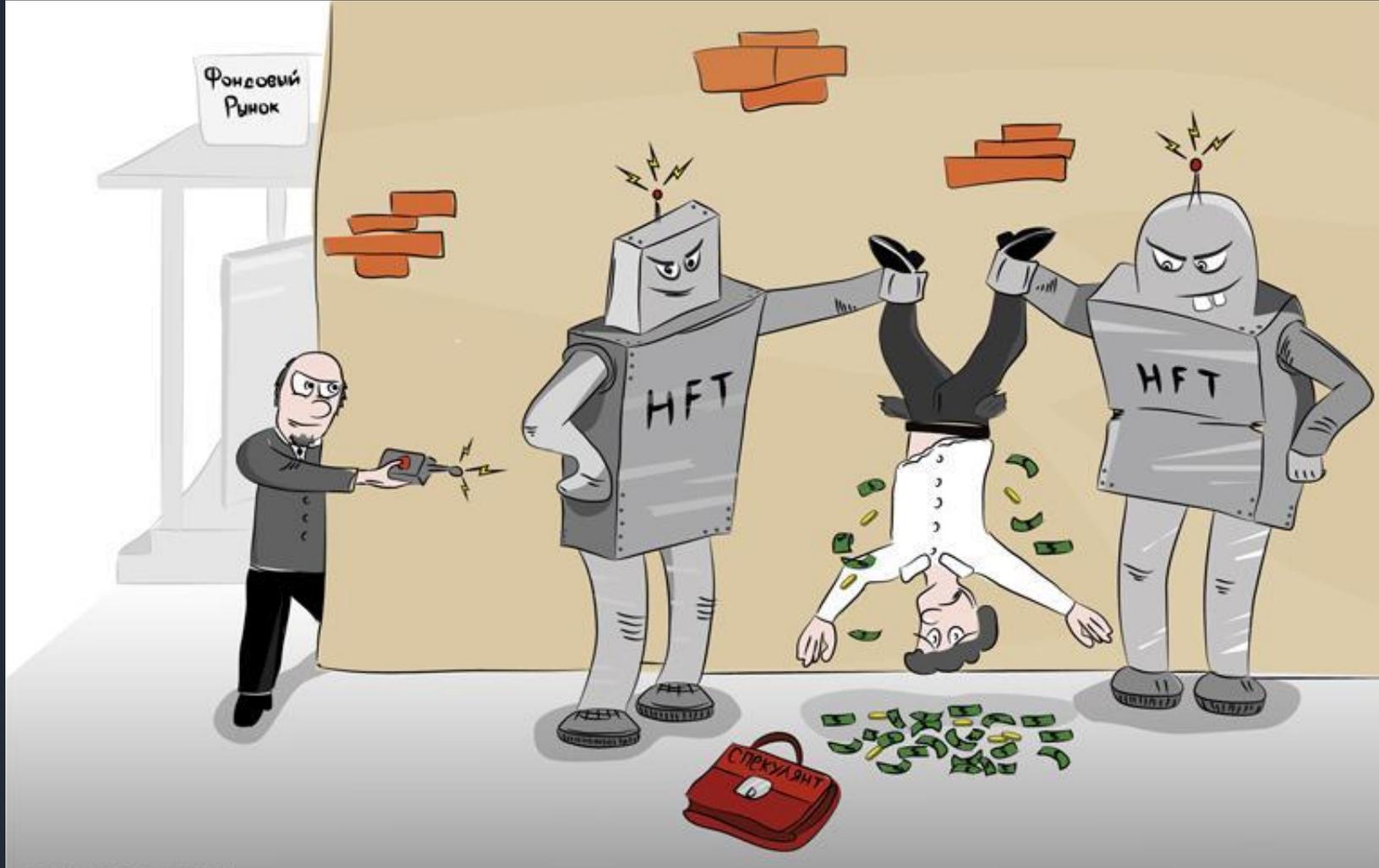




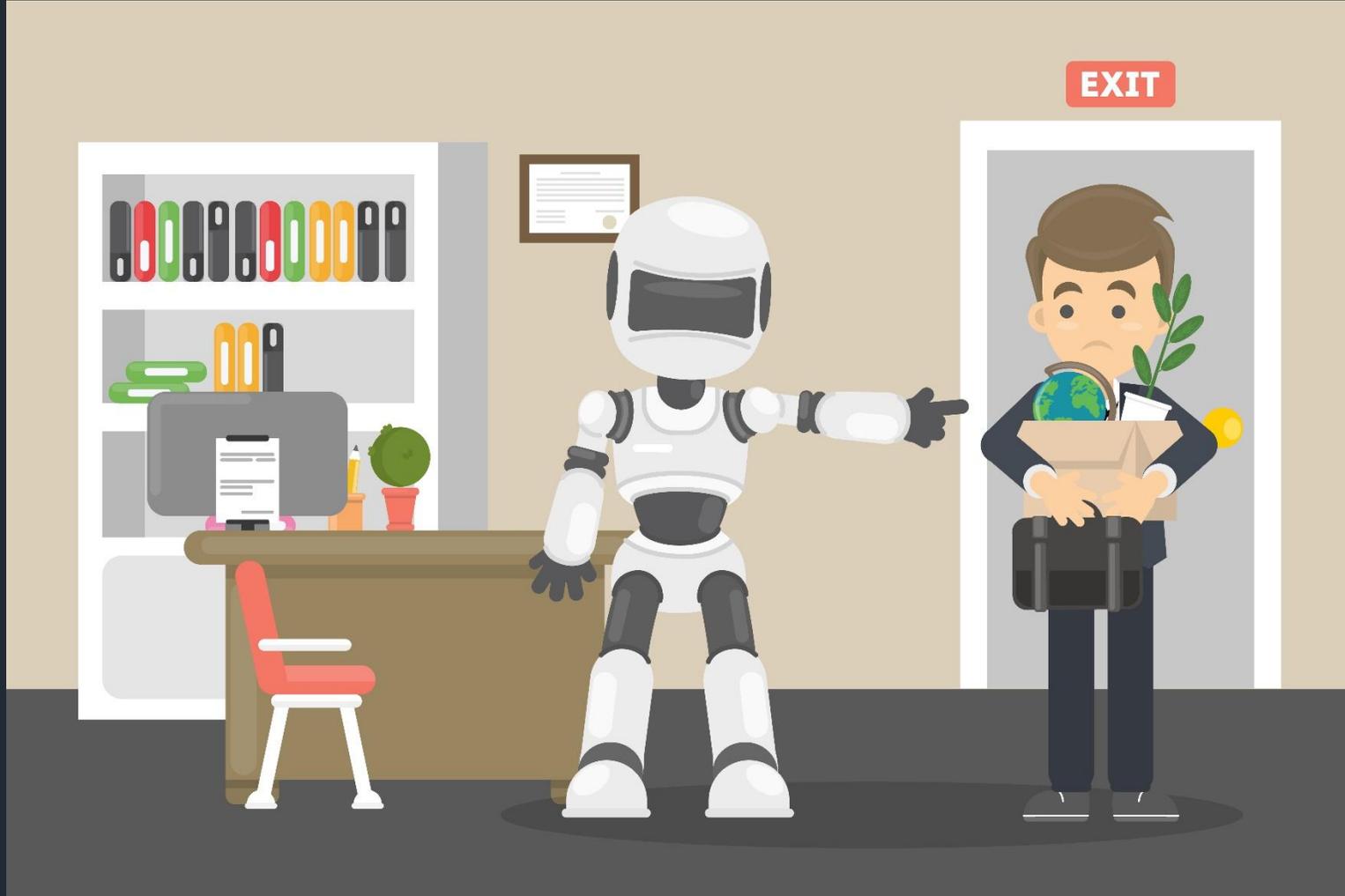
+ Масштаб



+ Экономия сил

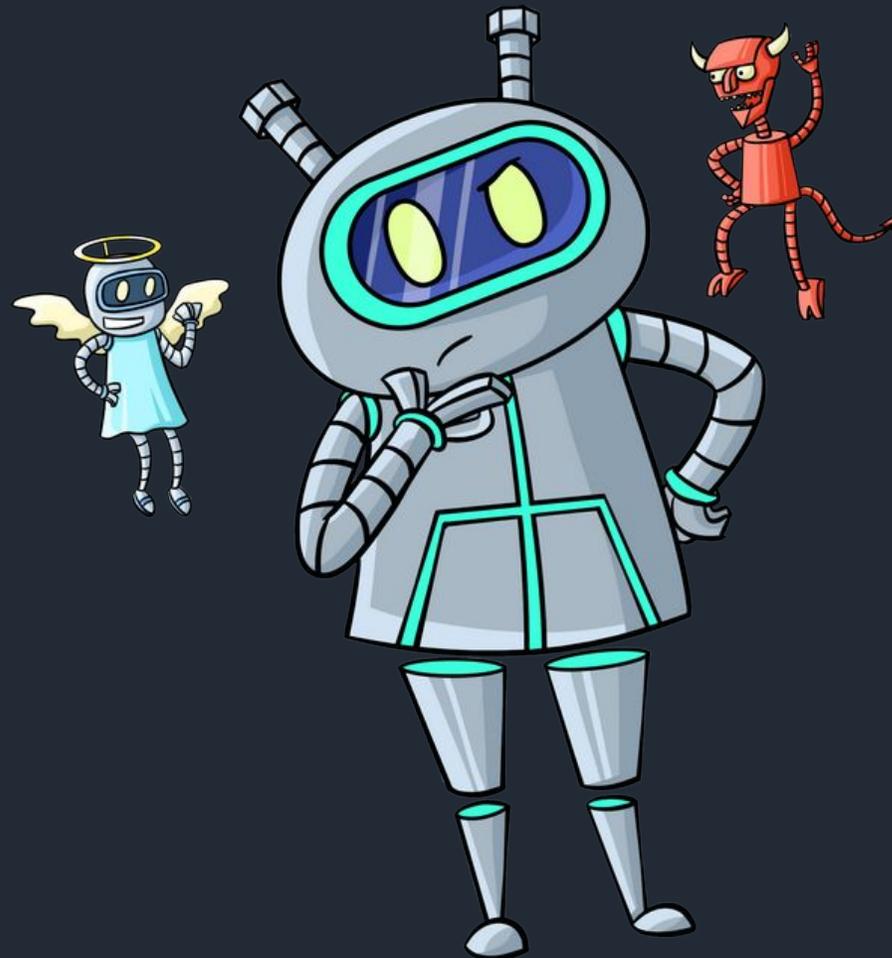


+ Экономия средств

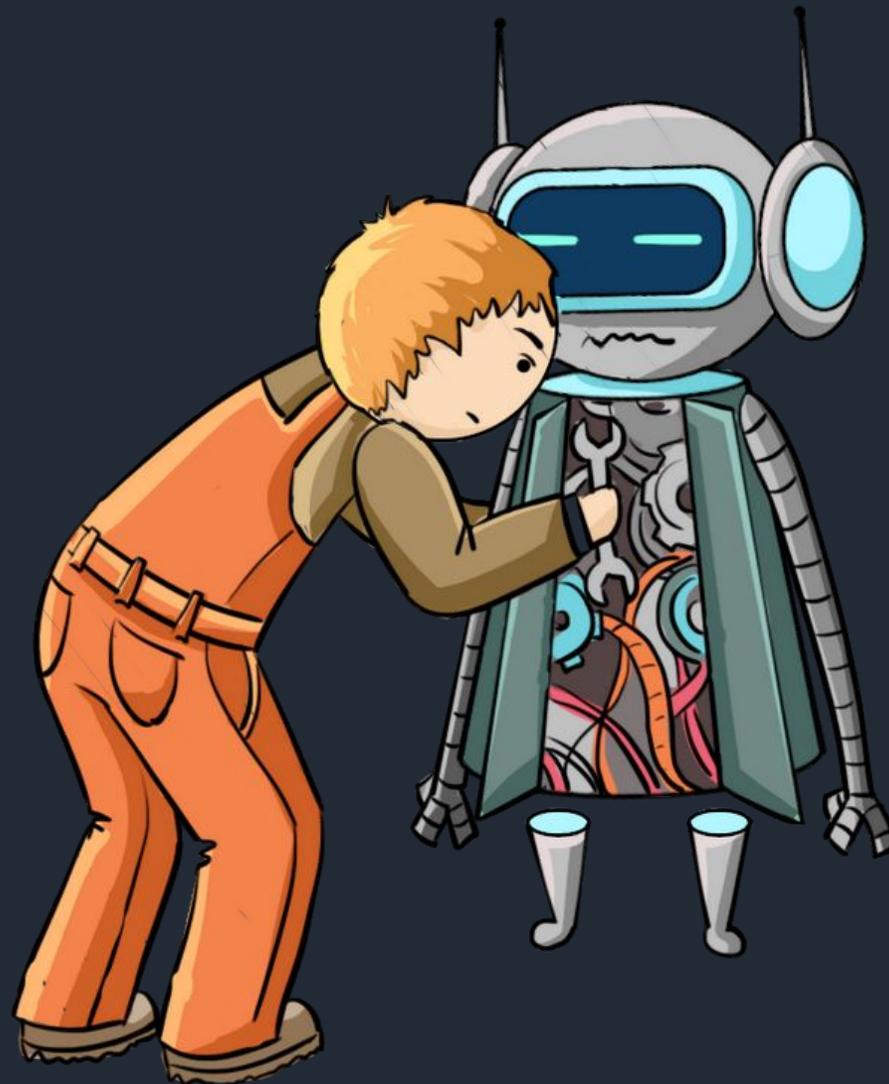


Минусы автоматизации

- Поломки
- Близорукость
- Трудно поддерживать
- Не везде применимы
- Затратность



- Поломки



- Близорукость



- Трудно



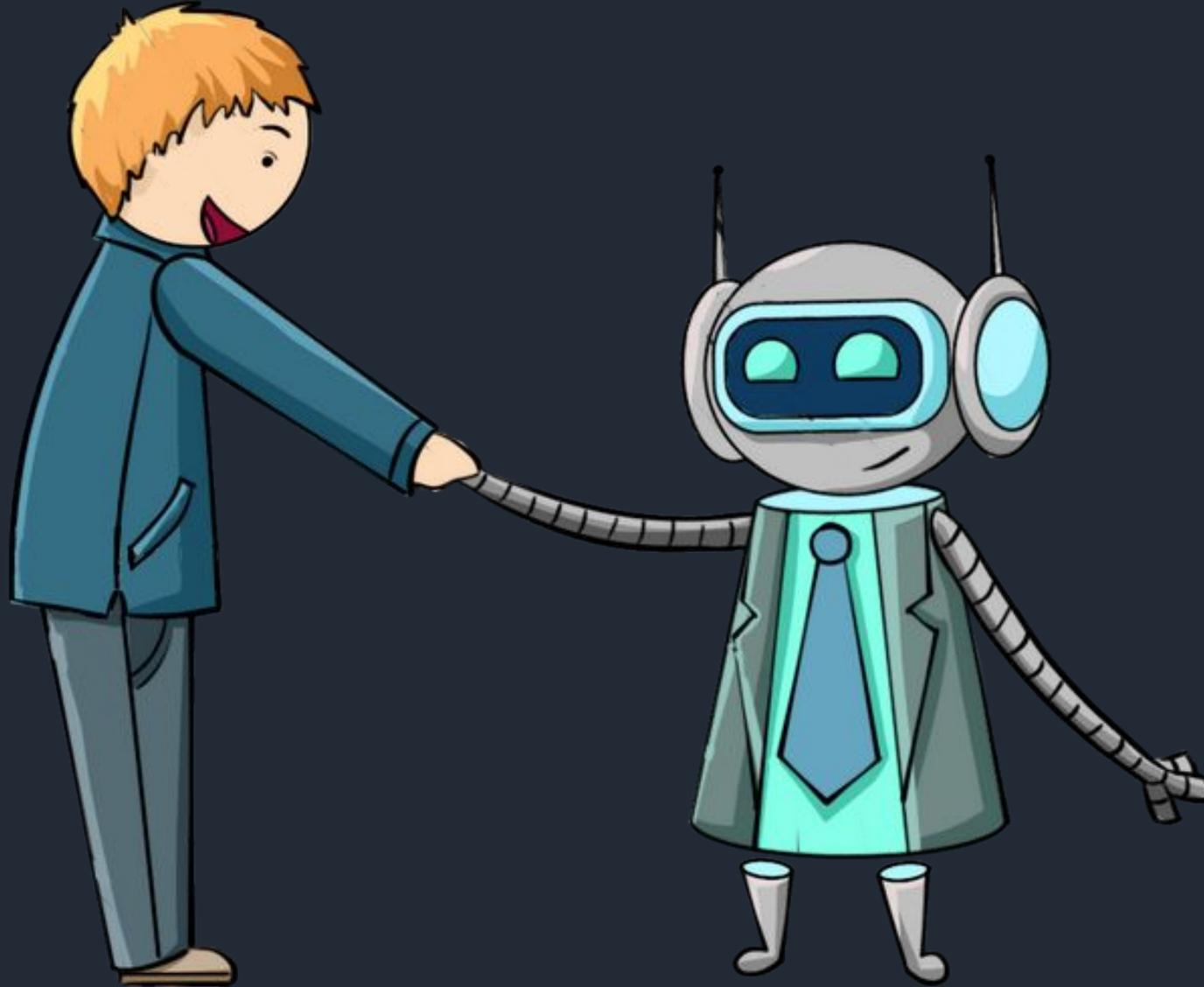
- Не везде



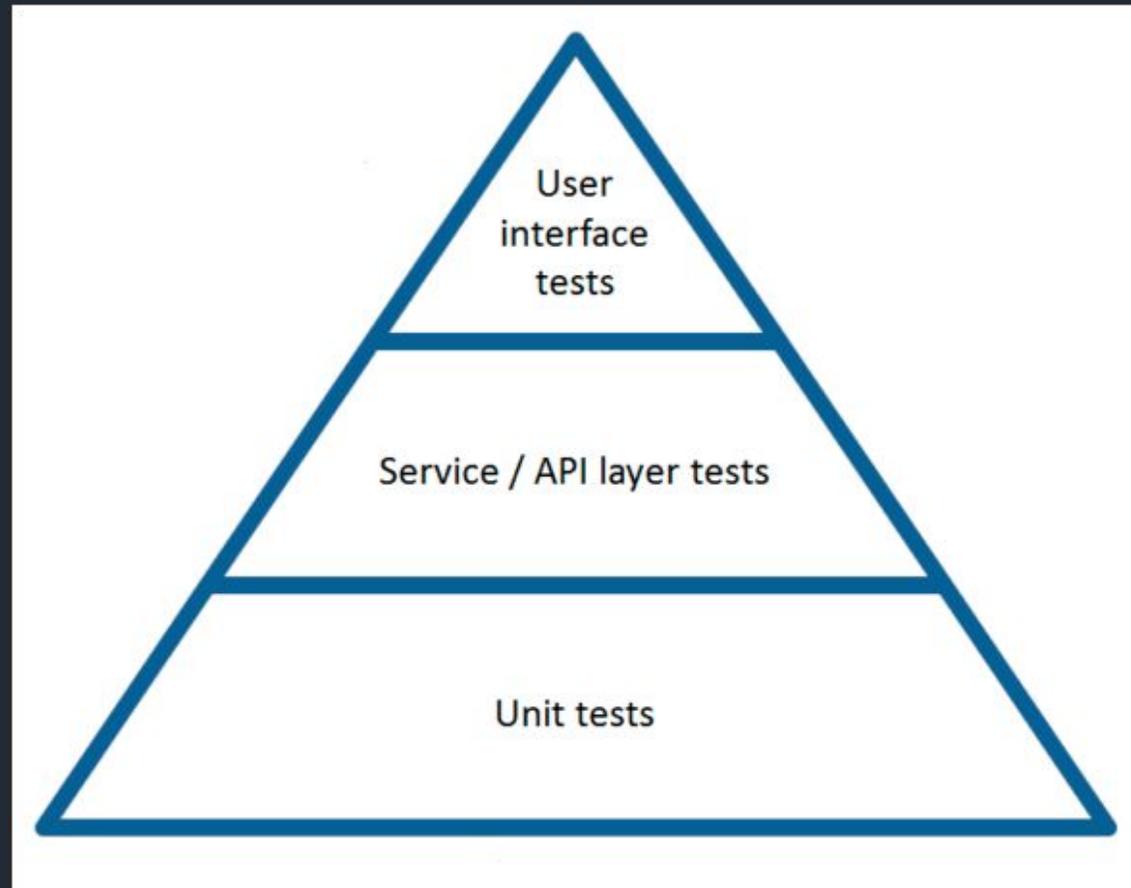
- Затратность



Выводы



Пирамида АТ



Когда автоматизировать?

- Regression testing**
- Performance testing
- Compatibility testing
- Unit testing
- Smoke testing
- Security testing
- GUI testing



Automated Regression Testing

Какую проблему решает автоматизация регрессионного тестирования?

- Необходимость выполнять ручные тесты, количество которых неуклонно растёт с каждым билдом, но вся суть которых сводится к проверке того факта, что ранее работавшая функциональность продолжает работать корректно.



Когда автоматизировать?

- Regression testing
- Performance testing**
- Compatibility testing
- Unit testing
- Smoke testing
- Security testing
- GUI testing



Automated Performance Testing

Какую проблему решает автоматизация производительности?

- Создание нагрузки с интенсивностью и точностью, недоступной человеку.
- Сбор с высокой скоростью большого набора параметров работы приложения.
- Анализ большого объёма данных из журналов работы системы автоматизации.



Когда автоматизировать?

- Regression testing
- Performance testing
- Compatibility testing**
- Unit testing
- Smoke testing
- Security testing
- GUI testing



Automated Compatibility Testing

Какую проблему решает автоматизация тестирования совместимости?

- Выполнение одних и тех же тест-кейсов на большом множестве входных данных, под разными платформами и в разных условиях.



Когда автоматизировать?

- Regression testing
- Performance testing
- Compatibility testing
- Unit testing**
- Smoke testing
- Security testing
- GUI testing



Automated Unit Testing

Какую проблему решает автоматизация модульного тестирования?

- Проверка корректности работы атомарных участков кода и элементарных взаимодействий таких участков кода — практически невыполнимая для человека задача при условии, что нужно выполнить тысячи таких проверок и нигде не ошибиться.



Когда автоматизировать?

- Regression testing
- Performance testing
- Compatibility testing
- Unit testing
- Smoke testing**
- Security testing
- GUI testing



Automated Smoke Testing

Какую проблему решает автоматизация smoke тестирования?

- Выполнение при получении каждого билда большого количества достаточно простых тест-кейсов необходимых для проверки работоспособности приложения.



Когда автоматизировать?

- Regression testing
- Performance testing
- Compatibility testing
- Unit testing
- Smoke testing
- Security testing**
- GUI testing



Automated Security Testing

Какую проблему решает автоматизация тестирования безопасности?

- Необходимость проверки прав доступа, паролей по умолчанию, открытых портов, уязвимостей текущих версий ПО и т. д., т. е. быстрое выполнения очень большого количества проверок, в процессе которого нельзя что-то пропустить, забыть или «не так понять».



Когда автоматизировать?

- Regression testing
- Performance testing
- Compatibility testing
- Unit testing
- Smoke testing
- Security testing
- GUI testing**



Automated GUI Testing

Какую проблему решает автоматизация тестирования пользовательского интерфейса?

- Автоматизация предельно рутинных действий (например, проверить покупку всех товаров в интернет-магазине).



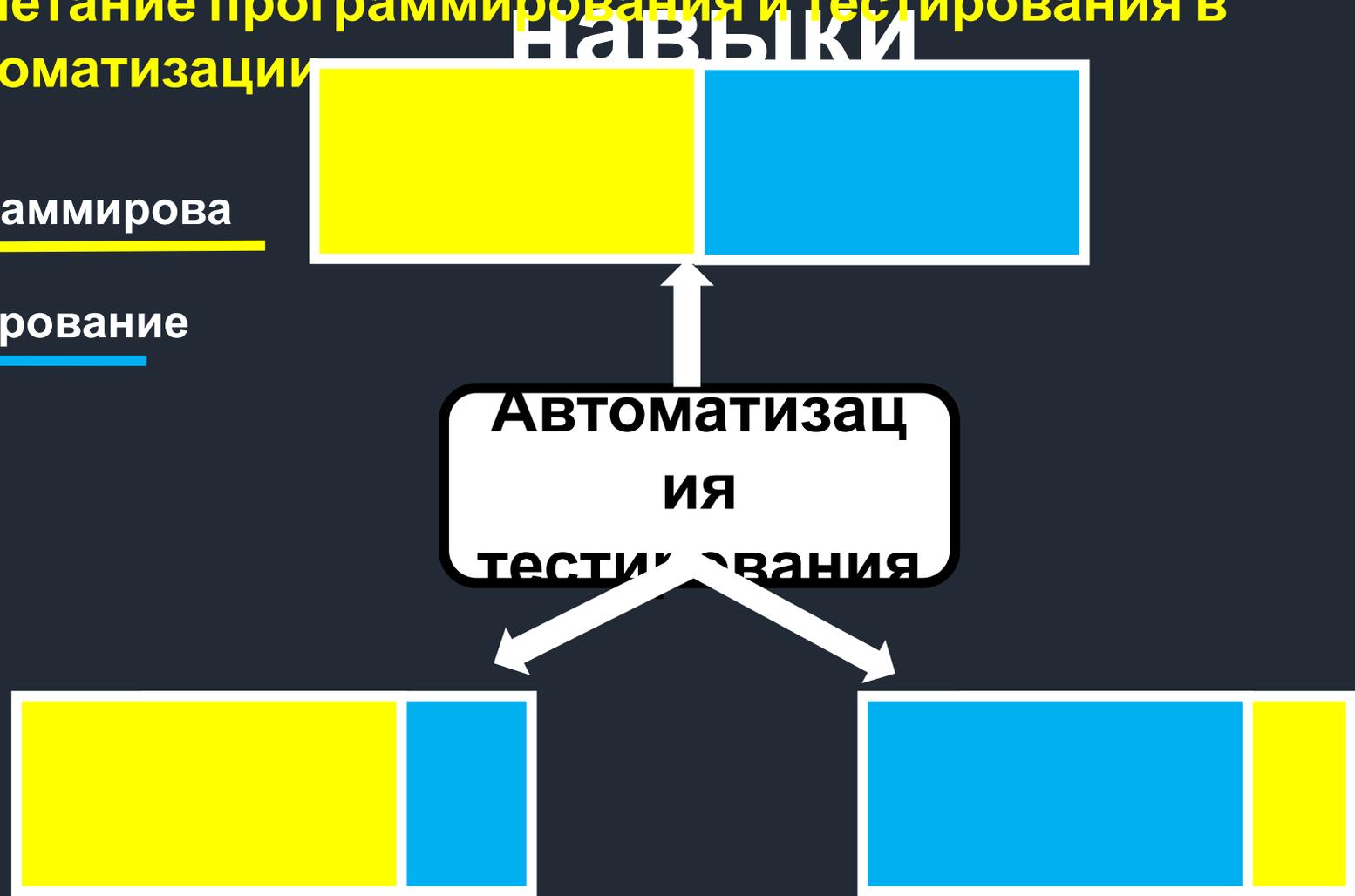
Выбор инструмента



Необходимые знания и навыки

Сочетание программирования и тестирования в автоматизации

- Программирование
- Тестирование



Внимание!!!



Спасибо за внимание

301-1.ru



СВЯЗЬ СО МНОЙ



- Телеграмм: [@korpusovv](https://t.me/korpusovv)
- Почта: korpusov.vd@yandex.ru

