

# Тестирование ПО. Урок 1

Введение в тестирования ПО

# Agenda

1. Что такое тестирование ПО?

---

2. Цели тестирования

---

3. Принципы тестирования

---

4. Уровни тестирования

---

5. Типы тестирования

---

6. Методы тестирования

---

7. Процесс тестирования ПО

---

8. Методологии разработки ПО

---

Что такое  
тестирование  
ПО?



# Определение тестирования

**Тестирование ПО** - это процесс исполнения программы или ее компонентов под воздействием определенных условий. Результаты тестирования записываются для дальнейшей оценки программы или ее компонентов и принятия решения о дальнейшей судьбе продукта.

**Тестирование ПО** - это процесс проверки, программы на предмет соответствия требованиям

# Почему тестирование важно?

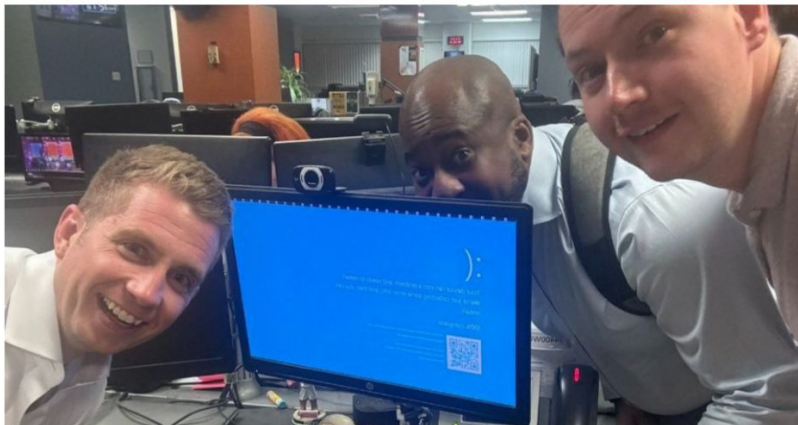
XStepanPetrov 19 июл 2024 в 15:08

Глобальный сбой в системах Microsoft поставил мир на грань техноапокалипсиса. Но почему в России все в порядке?

Простой 3 мин 48K

Блог компании Группа компаний X-Com, Будущее здесь

Мнение



used the IT Meltdown - The New York Times

Спросить Алис

The New York Times

## Chaos and Confusion: Tech Outage Causes Disruptions Worldwide

Airlines, hospitals and people's computers were affected after CrowdStrike, a cybersecurity company, sent out a flawed software update.

Listen to this article · 8:54 min Learn more Share full article 951



Travelers waiting to check in at the airport in Hamburg, Germany, on Friday. Bodo Marks/DPA, via Associated Press

By Adam Satariano, Paul Mozur, Kate Conger and Sheera Frenkel  
July 19, 2024

Airlines grounded flights. Operators of 911 lines could not respond

# Почему тестирование важно?

18:45, 2 июля 2013 Наука и техника



## Падение с переворотом «Протон-М» со спутниками на борту взорвался через 30 секунд после запуска

Алексей Тимошенко



Протон-М перед установкой. Верхняя часть ракеты с контейнером, внутри которого размещены спутники и разгонный блок. Фото: alexrpg / Flickr

2 июля на космодроме Байконур произошла авария: ракета-носитель «Протон-М» с тремя навигационными спутниками отклонилась от курса, загорелась и упала на первой же минуте полета. Помимо потери спутников происшествие обернулось тем, что в окружающую среду попали токсичное топливо и окислитель. Причины аварии пока неизвестны.

"Этот аварийный полет длился 32 секунды, - сказал Лопатин. - Авария произошла из-за потери стабилизации по каналу рыскания из-за нештатной работы датчиков угловых скоростей". По его словам, на 13-й секунде полета система управления ракеты-носителя сформировала команду превышения предельных значений углов, и была дана команда "авария". Тогда конструкция ракеты-носителя стала разрушаться в воздухе.

Специалисты подчеркивают, что все шесть датчиков до установки на борт ракеты-носителя прошли испытания без замечаний. А вот когда началась сборка...

"Причиной аварийного пуска ракеты космического назначения является нарушение технологии установки трех датчиков угловых скоростей по каналу рыскания на ракете-носителе "Протон-М" в ГКНПЦ имени Хруничева", - говорится в отчете о работе межведомственной комиссии по расследованию причин аварии ракеты, размещенном на сайте Роскосмоса. Отмечается, что применяемые способы контроля не позволяют выявить неправильную установку датчиков на ракете-носителе во время наземной подготовки и испытаний. "Дефект проявился в полете и имеет производственный характер", - подчеркивается в документе.

# Почему тестирование важно?

## Самый дорогой баг

Ошибка, которая привела к разрушению ракеты Ariane 5 в 1996 году, считается самым дорогим багом в истории, она стоила около 370 миллионов долларов. Через 37 секунд после старта ракета отклонилась от курса и самоуничтожилась. Причина — баг в программном обеспечении: при преобразовании 64-битного числа в 16-битное произошло переполнение буфера, что вызвало сбой в навигационной системе.

## Самый известный в поп-культуре

Самый известный баг из индустрии развлечений — это MissingNo. из франшизы про покемонов. Глюк появился в 1999 году в первых играх серии Pokemon Red и Pokemon Blue на приставке Game Boy. Баг возникал при выполнении трех определенных игровых действий и приводил к появлению странного покемона с искаженной графикой и необычными способностями. Всего существовало пять разных форм MissingNo.



# Почему тестирование важно?

## Безопасность



## Удобство



## Финансы

## Конкурентоспособность





Цели тестирования

2

# Цели тестирования

- Оценить качество сопутствующих документов: требований, дизайна, кода
- Убедиться в покрытии требований
- Убедиться, что продукт завершен и работает как ожидает клиент
- Предотвратить возникновение дефектов у пользователей
- Найти ошибки
- Предоставить клиенту, заказчику информацию о состоянии качества продукта
- Уменьшить риск неадекватного ПО
- Убедиться, что ПО соответствует требованиям государственных и юридических стандартов

# Принципы тестирования



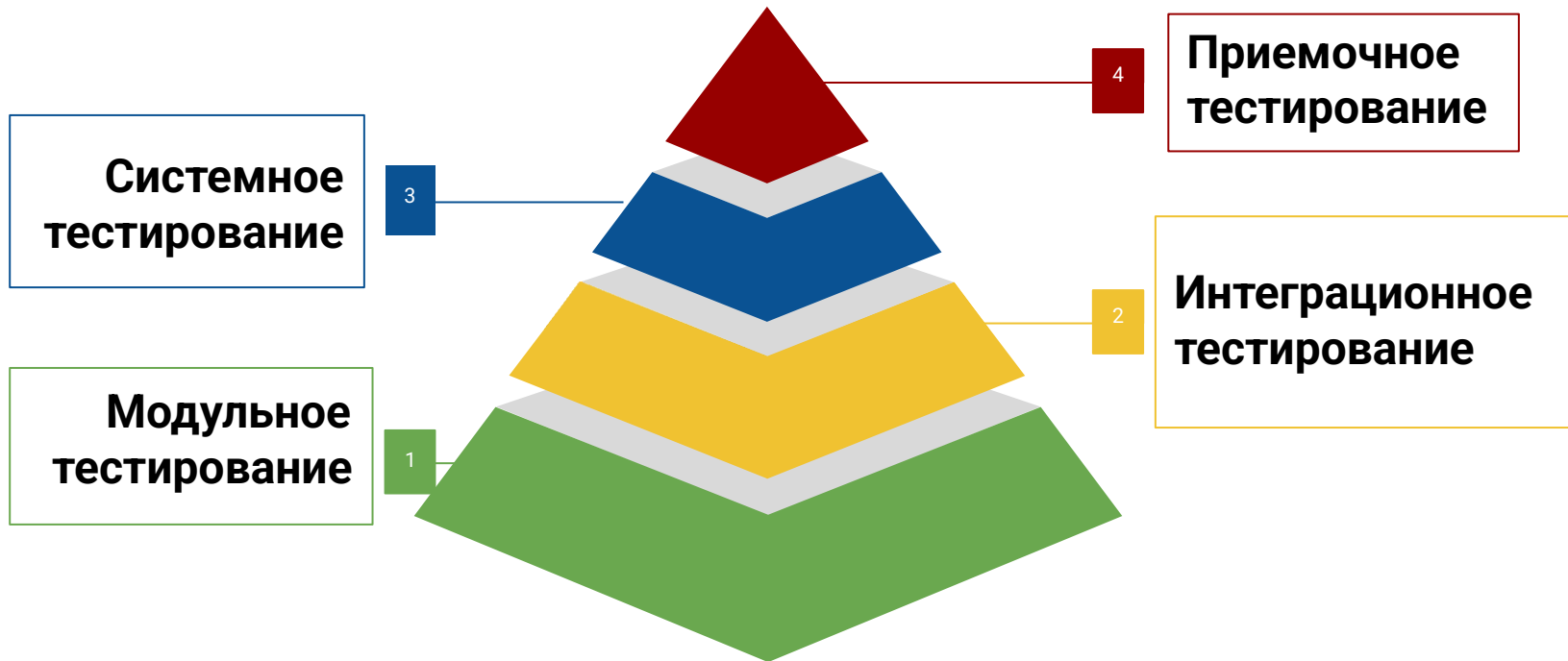
# Принципы тестирования

1. Тестирование показывает наличие дефектов, но не их отсутствие
2. Исчерпывающее (exhaustive) тестирование невозможно
3. Тестирование на ранних этапах может сохранить время и ресурсы
4. Дефекты имеют свойство скапливаться (defect clustering)
5. “парадокс пестицидов” или “pesticide paradox”
6. Тестирование контекстно зависимо
7. Нахождение и устранение всех ошибок не гарантирует качество

Уровни  
тестирования

4

# Уровни тестирования (Пирамида тестирования)



# Уровни тестирования

## Модульное тестирование (Unit testing)

**Unit testing, Модульное тестирование** (или Компонентное тестирование) - это тестирование отдельного взятого программного модуля, функции.

- Это самая ранняя стадия тестирования
- Unit тесты это всегда автоматические тесты
- Они могут покрывать отдельные функции, методы в коде или их совокупность
- Часто используют заглушки
- Запускаются при сборке или компиляции программы
- Как правило, пишутся разработчиками

# Уровни тестирования

## Интеграционное тестирование (Integration testing)

**Integration testing, Интеграционное тестирование** - это тестирование, в котором программные или аппаратные компоненты собраны воедино для проверки их взаимодействия (определение взято из стандарта IEEE 610.12- 1990).

- Может встречаться на разных фазах тестирования. Оно может быть как между компонентами(модулями) внутри одной системы, так и между системами
- Цель интеграционного тестирования протестировать взаимодействие между компонентами (модулями) или системами, их интеграцию друг с другом



# Уровни тестирования

## Системное тестирование (System testing)

**System testing, Системное тестирование** - это тестирование, которое проводится на полной, интегрированной системе для оценки соответствия продукта поставленным требованиям. В таком тестировании собраны все части(компоненты) системы вместе.

- Как правило это финальная фаза проверок
- Как правильно содержит много пользовательских сценарий (end-to-end)
- Должно выполняться на окружении близком к реальному окружению заказчика. Это необходимо, чтобы минимизировать риск пропустить ошибки, связанные с окружением

# Уровни тестирования

## Приемочное тестирование (Acceptance testing)

**Acceptance testing, Приемочное тестирование** - это формальная проверка, которая позволяет понять заказчику, или любому авторизованному лицу, принимать систему или нет.

- Цель такой проверки убедиться, что она выполняет поставленные цели, требования
- Зачастую такое тестирование выполняется заказчиком на окружении, близком к реальному
- Приёмочное тестирование может встречаться на всех уровнях проверки системы, например, для новой сборки программного обеспечения.

Типы тестирования

5

# Типы тестирования. Функциональное тестирование

## Функциональное тестирование (Functional testing)

– тестирование функций, которые приложение должно выполнять.

Может быть:

- **Позитивное**, направленное на проверку успешных сценариев, не подразумевающее ошибки пользователя
- **Негативное**, направленное на тестирование ошибок различного рода, которые пользователь может совершить и их обработку системой



# Типы тестирования. Нефункциональное тестирование

## Нефункциональное тестирование (Non-Functional testing)

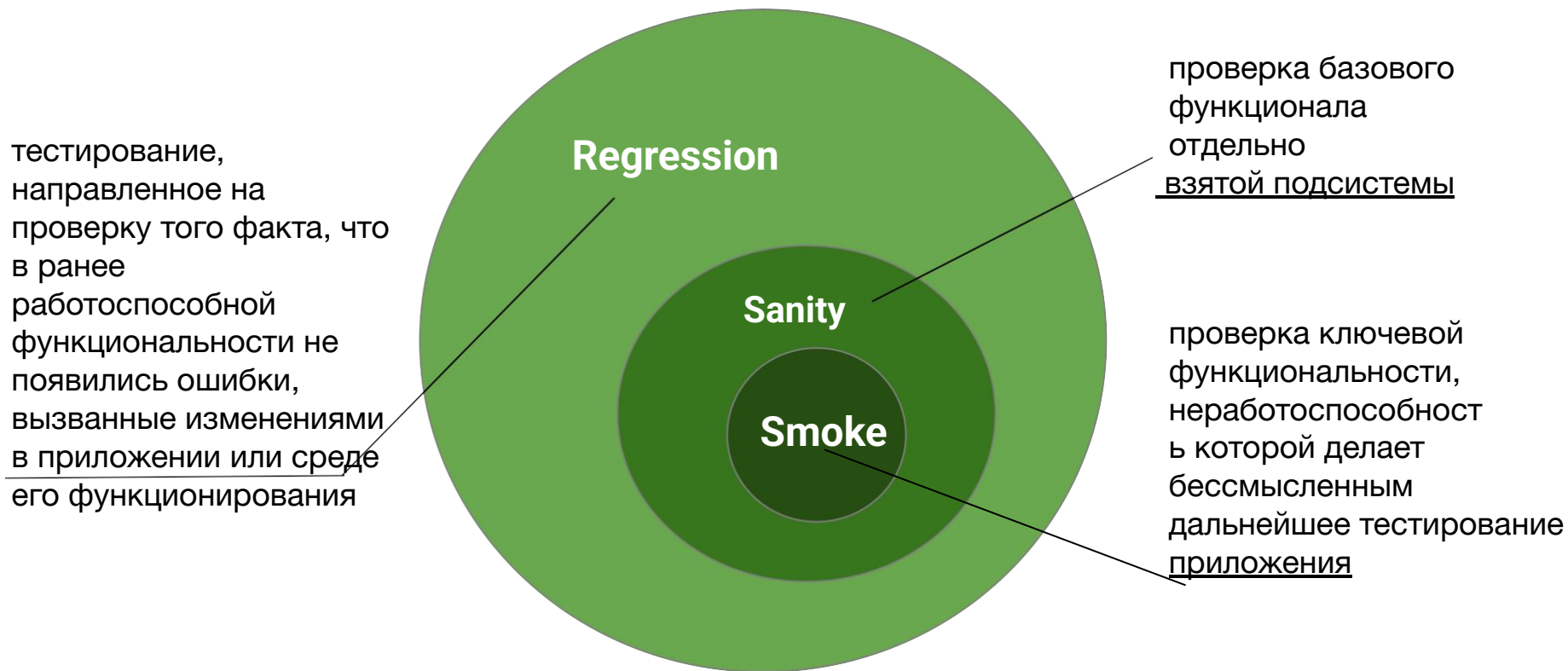
– тестирование всего, что не входит в функциональное тестирование, но является важным показателем качества системы.

Может быть:

- **Тестирование конфигурации** (Configuration, Maintenance, Supportability testing)
- **Тестирование надежности** (Reliability testing)
- **Тестирование инсталляции** (Installation, Deployment testing)
- **Тестирование безопасности** (Security testing)
- **Тестирование отказоустойчивости** (High Availability, Robustness testing)
- **Тестирование производительность** (Performance, Load, Longevity testing)
- **Стресс-тестирование** (Stress testing)
- **Тестирование удобства использования** (Usability testing)
- **Тестирование соответствия требованиям законодательств** (Compliance testing)



# Типы тестирования



Методы  
тестирования

6

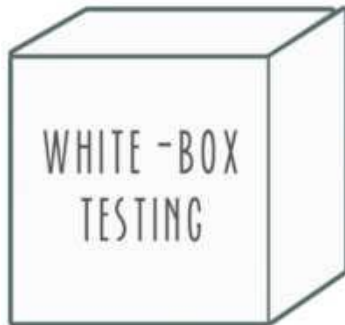
# Методы тестирования

## Метод черного ящика



Когда не знаем как  
устроена система  
внутри, проверяем  
интерфейсы

## Метод белого ящика



Когда знаем как  
устроена система  
внутри, используем эту  
информацию для  
тестирования

## Метод серого ящика



Комбинация белого и  
черного ящиков



Процесс  
тестирования  
ПО



# Процесс тестирования ПО

## 1. Планирование и review требований



## 2. Анализ и проектирование (дизайн)



## 3. Имплементация и исполнение



## 4. Завершение. Отчетность



**Вопросы?**

# Ресурсы

1. ISTQB Foundations,  
<https://www.istqb.org/downloads/send/51-ctfl2018/208-ctfl-2018-syllabus.html>
2. IEEE 610.12-1990,  
<http://www.mit.jyu.fi/ope/kurssit/TIES462/Materiaalit..>
3. QA fundamental terminology,  
<http://softwaretestingfundamentals.com/>
4. Святослав Куликов, Тестирование ПО. Базовый курс,  
[https://svyatoslav.biz/software\\_testing\\_book\\_download/](https://svyatoslav.biz/software_testing_book_download/)
5. Lee Copeland, A Practitioner's Guide to Software Test Design,  
<https://www.amazon.com/Practitioners-Guide-Software-Test-Design/dp/158053791X>
6. Статья по методологиям ПО,  
<https://habr.com/ru/company/edison/blog/269789/>