

# Вопросы:

1. Изобразите (как сможете 😊) ЖЦ дефекта.
2. Какие бывают резолюции дефектов?
3. Какие Вы знаете статусы дефектов?





# **ЛЕКЦИЯ 4**

## **«ТИПЫ И ВИДЫ ТЕСТИРОВАНИЯ»**



Тестирование инсталляции и лицензирование  
кроссплатформное  
Тестирование прототипа  
Валидация дефектов  
Тестирование требований  
Чёрный ящик  
Бета-тестирование  
Белый ящик  
Юзабилити  
По документации  
Доступности  
Интеграционные  
стрессово  
е

Интернационализация  
и  
Тестирование пользовательского интерфейса  
Автоматизированное  
Исследовательское  
Серый ящик  
Защищенность  
и  
Альфа-тестирование  
кроссбраузерное  
Тестирование новых функций  
Локализация  
и  
Приемное тестирование  
объемно  
е

Системное  
Регрессионное  
АТ  
Модульное  
Smoke  
Функциональное  
Ручное  
Безопасность  
и  
Smoke  
нагрузочно  
е

Модуль  
о

Мат

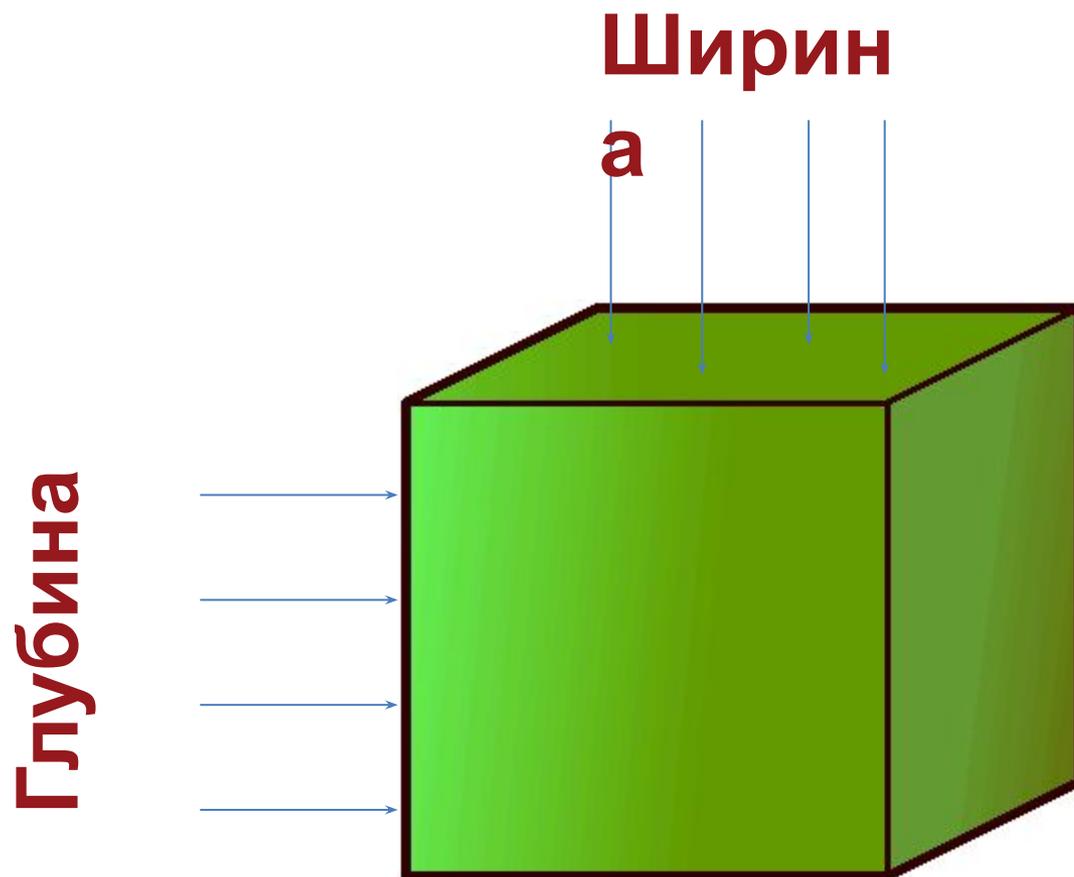
Интуитивное

## **Тестовое покрытие –**

это одна из метрик оценки качества тестирования, представляющая из себя плотность покрытия тестами требований либо исполняемого кода

## **Тестовое покрытие** определяется:

- Глубиной тестирования
- Шириной тестирования
- Целями тестирования
- Профессиональным уровнем тестировщика



## **Ширина тестирования**

отражает количество модулей/функций приложения, которые были протестированы.

## **Глубина тестирования**

отражает вид проверок, которые производились для модуля/функции (например, позитивность/негативность сценариев, тестирование в рамках/вне рамок тест сценариев и др.).

# По уровню покрытия



- **Smoke** – пригодность сборки для дальнейших тестов

Уровень качества: **Acceptable** / **Unacceptable**

- **MAT (Minimal Acceptance Test, Positive test)** – тест на положительных сценариях

Уровень качества: **High** / **Medium** / **Low**

- **AT (Acceptance Test)** – полный тест на положительных и негативных сценариях)

Уровень качества: **High** / **Medium** / **Low**

# По цели тестирования



- **New Feature Test (NFT, AT of NF)** – определить качество поставленной на тестирование функциональности
- Обычно тестируется полным тестом (AT)

# По цели тестирования



- **Regression testing** – удостовериться, что добавление новой функциональности не повлияло на работу старой
- Регрессионное тестирование может быть проведено на уровне *Smoke*, *MAT* или *AT*.

# По цели тестирования



- **Defect Validation** – проверка результата исправления дефектов
- Может включать элементы регрессионного тестирования.

# По цели тестирования



- **User Acceptance Test (UAT, ПиМИ):** формальная проверка приложения (зачастую заказчиком), сопровождаемая протоколированием результатов

## Тестовые активности:

- Smoke
- Регрессионное
- Валидация дефектов
- Тестирование новых функций

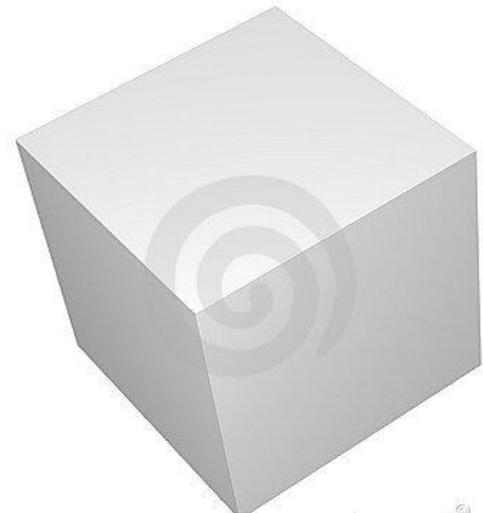
# По степени автоматизации



- **Автоматизированное**
- **Ручное**

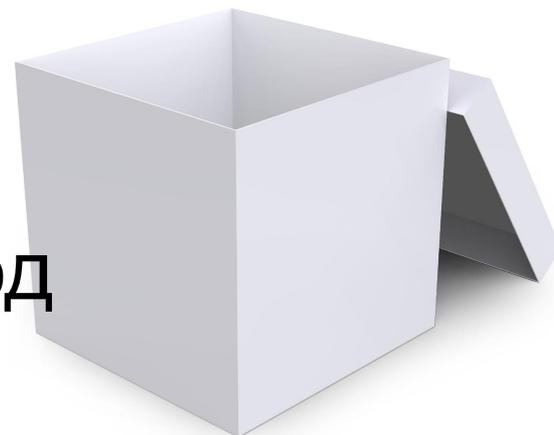
# По знанию кода

- **Черный ящик: не заглядываем внутрь системы**



# По знанию кода

- **Черный ящик:** не заглядываем внутрь системы
- **Белый ящик:** смотрим код



# По знанию кода

- **Черный ящик:** не заглядываем внутрь системы
- **Белый ящик:** смотрим код
- **Серый ящик:** частично заглядываем в систему

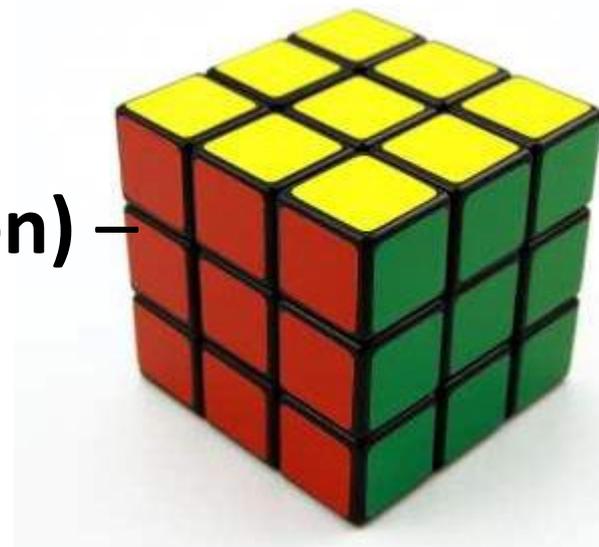


# По уровням

**Модульное (Unit/component)** –  
отдельные части (модули)  
СИСТЕМЫ

**Интеграционное (Integration)** –  
взаимодействие между  
отдельными модулями

**Системное (System)** –  
работоспособность системы в  
целом



# По подготовленности



- Интуитивное (Ad Hoc)
- Исследовательское (Exploratory)
- По документации

# По времени и месту проведения



- **Альфа- тестирование**
- **Бета- тестирование**

# По объекту



- Функциональное
- Тестирование пользовательского интерфейса
- Юзабилити
- Доступности
- Локализации
- Интернационализации
- Совместимости (кроссбраузерное, кроссплатформенное)
- Производительности (нагрузочное, стрессовое, объемное)
- Безопасности
- Защищенности
- Тестирование требований
- Тестирование прототипа
- Тестирование инсталляции и лицензирование

**Функциональное тестирование (Functional testing)** – это тестирование функций приложения на соответствие требованиям.

Оценка производится на основе сопоставления ожидаемых и полученных результатов (указанных в функциональной документации).

Важным условием является ввод корректных и некорректных тестовых

**Тестирование графического интерфейса пользователя (GUI)** предполагает проверку соответствия приложения требованиям к графическому интерфейсу, профессионально ли оно выглядит, выполнено ли оно в едином стиле.

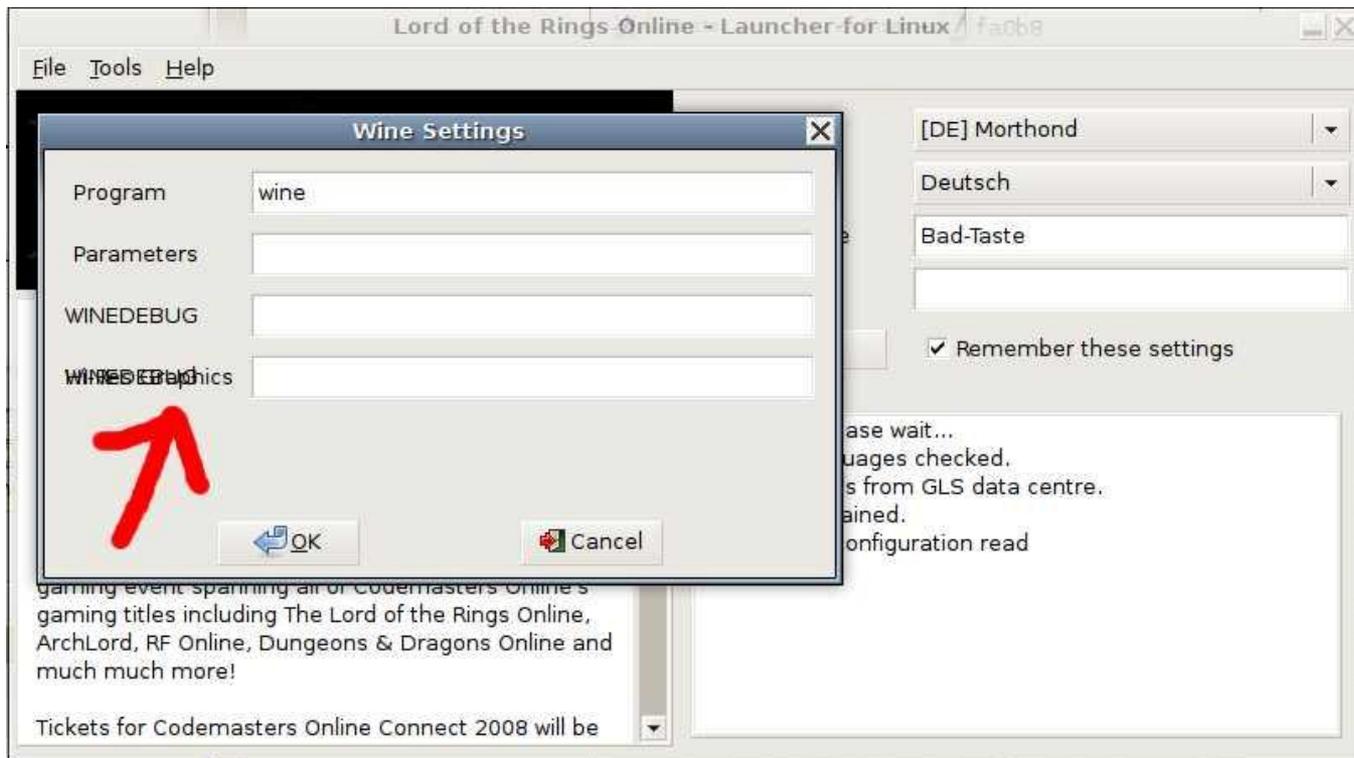
Как правило, через пользовательский интерфейс и реализуется большая часть функциональности ПО.

## Что нужно проверить?

- Горячие клавиши (hot keys)
- Порядок табуляции
- Окна сообщений
- Изменение размеров окон приложения
- Поддержка скроллинга

# Тестирование графического интерфейса пользователя

## Пример



Тестирование программ установки и лицензирования  
тестирование

В ходе **тестирования установки** (**Installation testing**) проверяется правильность процесса установки приложения и эффект, оказываемый установкой приложения на операционную систему конечного пользователя.

**Тестирование лицензирования** помогает определить эффективность системы защиты приложения **механизма**

## Характеристики требований

- Единичность
- Завершённость
- Последовательность
- Атомарность
- Актуальность
- Выполнимость
- Обязательность
- Проверяемость

Developer V/s Tester



## **Кроссбраузерность (Crossbrowser testing):**

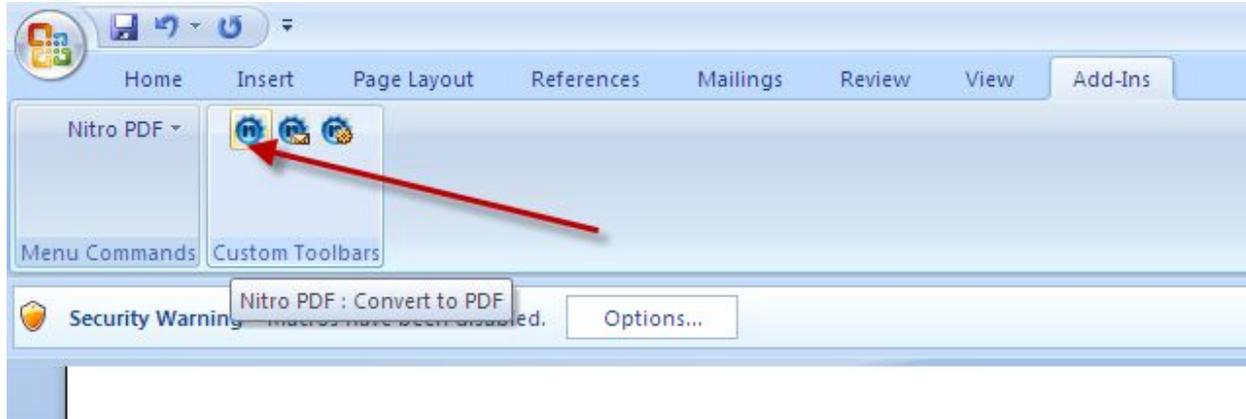
Цель данного типа тестирования – удостовериться, что приложение будет эффективно работать не только с Internet Explorer, но также и с альтернативными браузерами (Safari, Opera, FireFox, Google Chrome и др.)

## **Кроссплатформенность (Crossplatform testing):**

Цель кроссплатформенного тестирования – удостовериться, что приложение работает одинаково корректно и эффективно на любых типах и версиях операционных систем (ОС - Windows, Mac

**Тестирование интеграций (Integration testing)** – проверка того, как одно приложение взаимодействует с другим.

**Тестирование интеграций** – одна из фаз тестирования, при которой отдельные программные модули объединяются и тестируется их совместная работа.



### **Возможные проблемы:**

- дефекты самого продукта
- дефекты приложения, с которым происходит взаимодействие
- дефекты, возникающие в результате интеграции приложений

### Тестирование безопасности (Security testing)

представляет собой ряд действий, от разработки политики безопасности до тестирования на уровне приложения, операционной системы и сетевой безопасности.

В общем случае включает внедрение скриптов, операторов SQL, манипуляции с адресной строкой и исходным кодом программы

Modify item: **admin**

User name:*	<input type="text" value="admin1"/>
Password:*	<input type="text" value="admin1"/>

\*\*\*\*\*



**Удобство пользования (Usability)** – это степень эффективности, продуктивности и удовлетворенности, с которой продукт может использоваться определенными пользователями для достижения своих целей в определенном контексте (ISO 9241)

**Тестирование удобства пользования (Usability testing)** определяет, насколько комфортна работа пользователей с приложением или сайтом в контексте понимания процесса работы, простоту навигации и структуры, понятность и последовательность сообщений. Этот вид тестирования в огромной степени определяет будущий успех или провал проекта.

## **Критерии тестирования удобства пользования:**

- Функциональность
- Визуальное оформление
- Эффективность
- Производительность
- Обучаемость
- Навигация
- Логичность
- Обратная связь

## **Тестирование доступности (accessibility testing)**

необходимо, чтобы проверить соответствие программного обеспечения общепризнанным стандартам доступности. Практически все современные приложения, рассчитанные на широкую целевую аудиторию, должны быть доступны для использования людьми с ограниченными возможностями (зрительными, слуховыми, речевыми, когнитивными, неврологическими и др.), с учетом возраста, технических предпочтений и физических способностей.

## **Критерии тестирования доступности:**

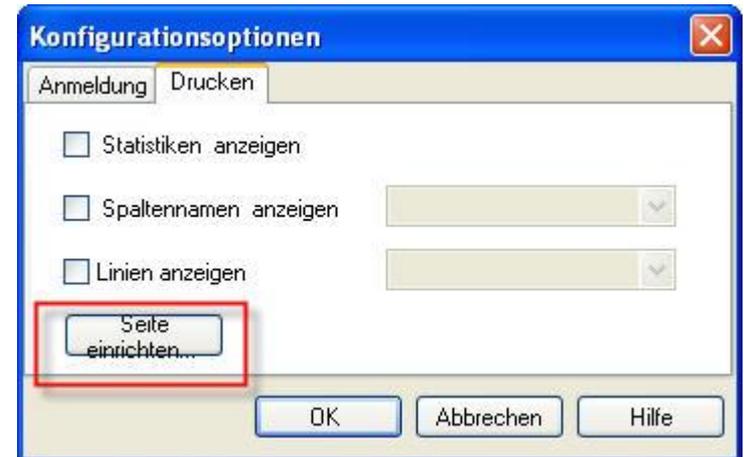
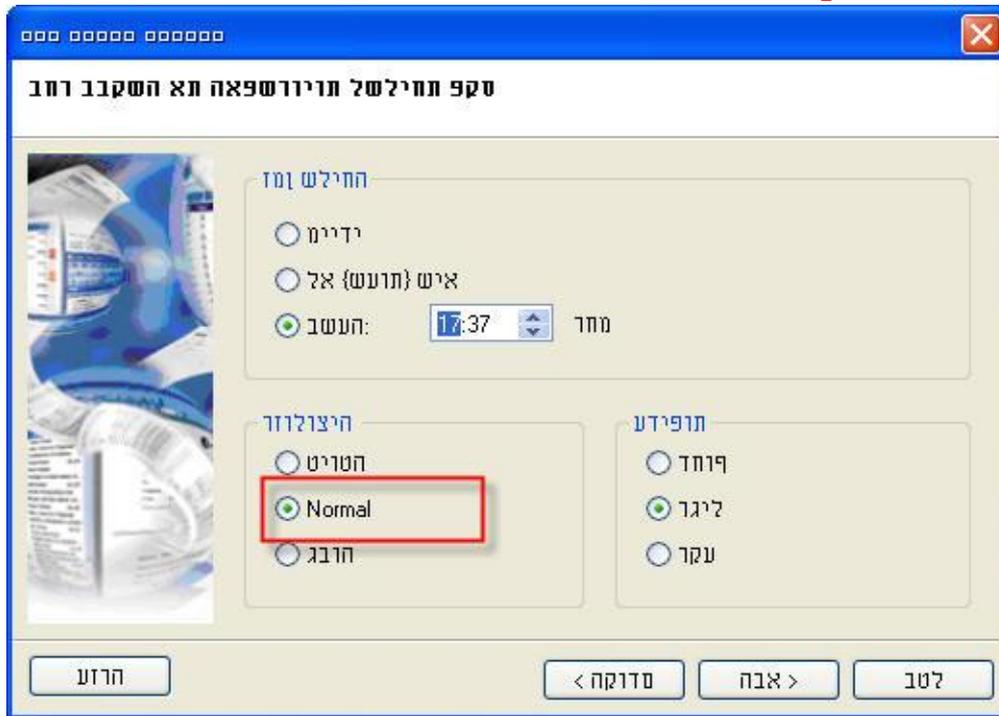
- Все изображения на сайте должны иметь альтернативный текст
- Аудио содержимое должно иметь текстовый эквивалент
- Страница должна быть читабельна при увеличенном размере шрифтов
- Страница должна читаться на монохромном экране и др.

Целью **тестирования локализации (Localization testing, L10n)** является проверка соответствия приложения региональным стандартам пользовательского интерфейса, языковым стандартам и др.

**В ходе тестирования локализации и интернационализации проверяются следующие аспекты:**

- Соответствие приложения стандартам оформления пользовательского интерфейса
- Отсутствие непереуведенных сообщений
- Соответствие способа подачи данных региональным стандартам
- Сортировка данных в алфавитном порядке
- Совместимость приложения со стандартами различных регионов
- Правильность оформления документации и вспомогательных файлов

# Примеры дефектов



**Тестирование интернационализации**  
**(Internationalization testing, i18n)** – проверка того, каким образом происходит адаптация продукта.

**Интернационализация (англ. internationalization)** — технологические приёмы разработки, упрощающие адаптацию продукта (такого как программное или аппаратное обеспечение) к языковым и культурным особенностям региона (регионов), отличного от того, в котором разрабатывался продукт.

**Тестирование производительности (Performance testing)** — тестирование, которое проводится с целью определения, как быстро работает вычислительная система или её часть под определённой нагрузкой.

В тестировании производительности различают следующие направления:

- нагрузочное (load)
- стресс (stress)
- объемное (volume)

## Задача 1

- Вам предоставляется на тестирование веб-проект – интернет магазин по продаже деталей для автомобилей.
- Из планов разработки известно, что на тестирование будут поставлены **3 сборки**:
  - #1 включает в себя 50% функциональности
  - #2 включает в себя оставшиеся 50% функциональности
  - #3 является финальной (release candidate) и будет отдана заказчику
- **Вопрос:**
  - Какие тесты Вы проведете на каждой сборке?

**Спасибо за  
внимание! Жду  
Ваших вопросов**

