

Тема 5

«Принципы разработки тестов»

ИЛИ

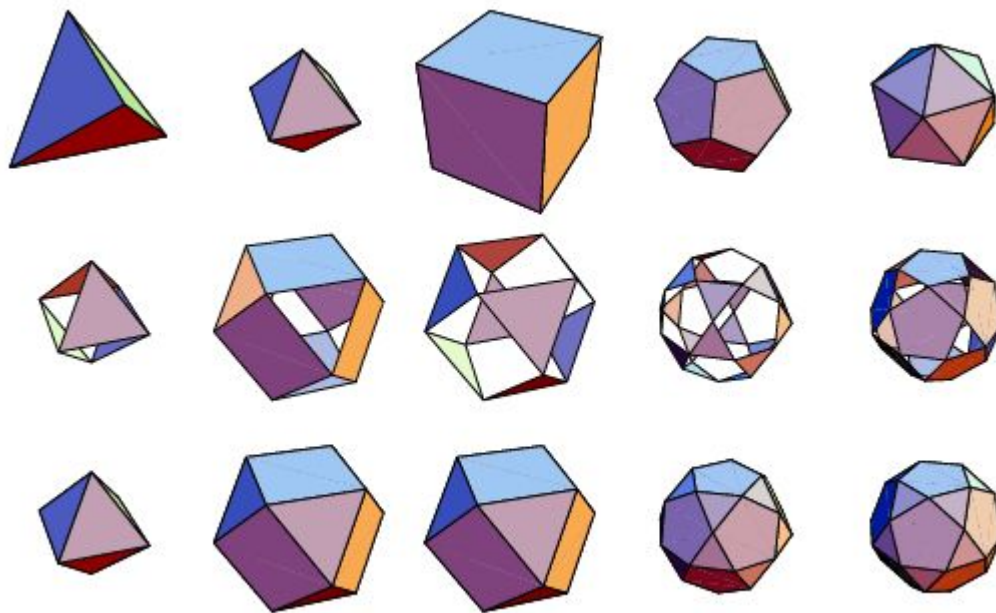
«как покормить кошку»

Что мы изучили в прошлой теме? Что запомнили?

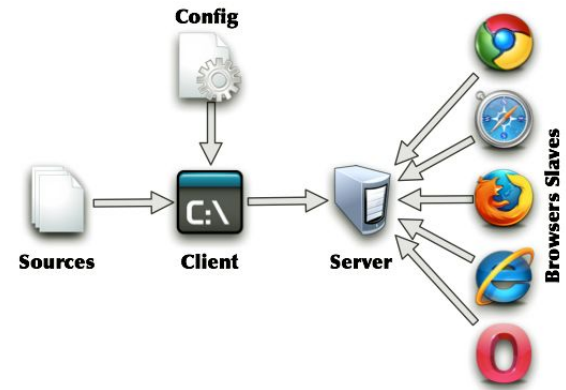
- ✓ Пути выявления требований.
- ✓ Особенности анкетирования.
- ✓ Направления прототипирования.
- ✓ Что было легче всего?
- ✓ Что вызвало сложности?

Виды тестов

+ вспомним пару определений



Тест-кейс (test case) – это набор **входных данных**, **условий выполнения** и **ожидаемых результатов**, разработанный с **целью** проверки того или иного свойства или поведения программного средства.



Потренируемся на
кошках!

Тест-кейс

Тест – «триплет» Вход/Состояние/Выход – последовательность шагов/действий, которая переводит систему из одного состояния в другое

Триплет **ISO**, где:

- [**I**] – is input data or action (входные данные или действия)
- [**S**] – is State of system at which data will be input (состояние системы, которая получает входные данные или воздействие)
- [**O**] – is the expected Output (ожидаемые Выход, выходные данные или выходной состояние системы)

Выполненный с определенной Целью!



Чек-лист (check-list) – это набор **идей тестов**.



Задание (см. следующий слайд)

```
root@vpupkin# cat /dev/ass > /dev/head
```

30.03.12 14:25 #AA-217272

Arris: Система

восстановления пароля

говорит, что пользователя

с таким емейлом у нее

нет. Но и зарегаться

заново не позволяет,

говорит — «пользователь

с таким емейлом уже

зарегистрирован»

фейспалм

Подписаться на рассылку

Подписаться



Какие бывают тесты

Основные **виды тестов**:

- позитивные;
- негативные.

Направления тестирования:

- статическое;
- динамическое.

Методы тестирования:

- чёрный ящик;
- белый ящик;
- серый ящик.

Виды тестирования:

- инсталляционное;
- регрессионное;
- нового функционала;
- конфигурационное;
- совместимости;
- удобство использования;
- интернационализации;
- локализации;
- исследовательское.

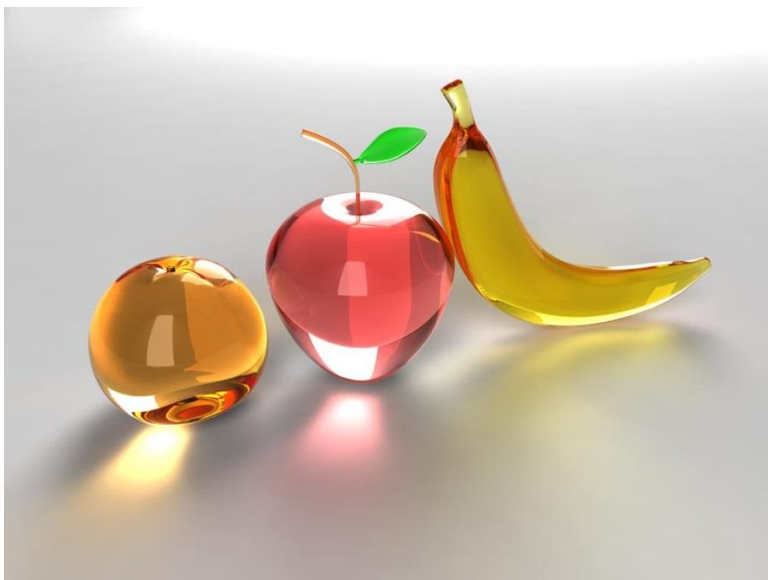
Подписаться на рассылку

Подписаться



Классы эквивалентности

Как сэкономить уйму времени!



Класс эквивалентности (equivalence class) – набор тестов, полное выполнение которого является избыточным и не приводит к обнаружению новых дефектов.



По всем пунктам – активно обсуждаем!

Признаки эквивалентности (несколько тестов эквивалентны, если):

- Они направлены на поиск одной и той же ошибки.
 - Если один из тестов обнаруживает ошибку, другие её тоже, скорее всего, обнаружат.
 - Если один из тестов **НЕ** обнаруживает ошибку, другие её тоже, скорее всего, **НЕ** обнаружат.
- Тесты используют схожие наборы входных данных.
 - Для выполнения тестов мы совершаем одни и те же операции.
 - Тесты генерируют одинаковые выходные данные или приводят приложение в одно и то же состояние.
- Все тесты приводят к срабатыванию одного и того же блока обработки ошибок («error handling block»).
 - Ни один из тестов не приводит к срабатыванию блока обработки ошибок («error handling block»).

Граничные условия
(border conditions) – это те
места, в которых один класс
эквивалентности переходит
в другой.



Граничные условия очень
важны, и их обязательно
следует проверять в тестах, т.
к. именно в этом месте чаще
всего и обнаруживаются
ошибки.



DTD&C.pdf

kanerPadmanab
ACONDomainTes

Пример

Проверить реакцию приложения на ввод слишком короткого (менее трёх символов) или слишком длинного (более 20-ти символов) имени пользователя, которое может содержать только английские буквы, цифры и знак подчёркивания.

Классы эквивалентности:

- 1. Позитивный тест:** строка допустимых символов длиной от трёх до 20-ти символов включительно.
- 2. Негативный тест:** строка короче трёх символов.
- 3. Негативный тест:** строка длиннее 20-ти символов.
- 4. Негативный тест:** строка длиной от трёх до 20-ти символов, содержащая недопустимые символы.

Тесты:

1. ABCD, ABCDEFGHIJKLMNOPQRST, abc_12_DEF
2. AA, {пустая строка}
3. AAAAAAAAAAAAAAAAAAAAAAAAAA (21 символ)
4. Abcd#23456%#@#&#%^#

Все ли ?
правильно

И ещё один пример. Для обсуждения!

«Чтобы добавить файл в свою фотогалерею на сайте, пользователь должен кликнуть по кнопке Открыть, выбрать файл и кликнуть по кнопке ОК».

Давайте абстрагируемся от пользовательского интерфейса и подумаем о файле. **Какие случаи нам надо будет проверить?**

Выводы

- **Классы эквивалентности не всегда очевидны.**
- Как правило, **негативных тестов** получается больше, чем **позитивных**.
- **Принадлежность теста к позитивным или негативным зависит от требований.**



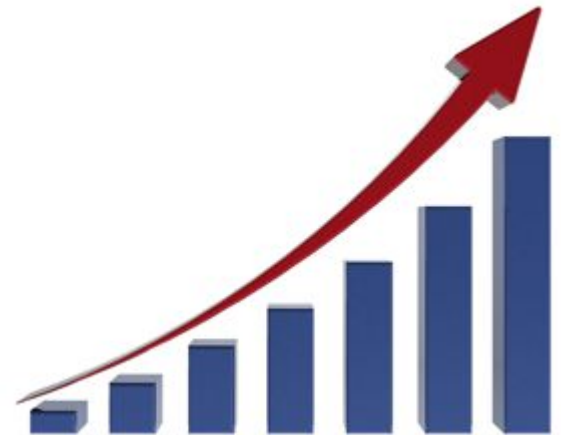
Рекомендации по разработке тестов

- Начинаяте с **простых очевидных тестов.**
- Затем **переходите к более сложным тестам.**
- Помните о **граничных условиях.**
- Если остаётся время, занимайтесь **исследовательским тестированием.**



Последовательность разработки и выполнения тестов

- Простые позитивные.
- Простые негативные.
- Сложные позитивные.
- Сложные негативные.



Документирование тестов

Приоритет

Связанное с тестом требование

Заглавие (суть) теста

Ожидаемый результат по каждому шагу

UG_U 1.12	A	R97	Галерея	Загрузка файла	<p>Галерея, загрузка файла, имя со спецсимволами</p> <p>Приготовление: создать непустой файл с именем #\$\$%^&.jpg</p> <ol style="list-style-type: none"> 1. Нажать кнопку «Загрузить картинку» 2. Нажать кнопку «Выбрать» 3. Выбрать из списка подготовленный файл 4. Нажать кнопку «ОК» 5. Нажать кнопку «Добавить в галерею» 	<p>1. Появляется окно загрузки картинки</p> <p>2. Появляется диалоговое окно браузера выбора файла для загрузки</p> <p>3. Имя выбранного файла появляется в поле «Файл»</p> <p>4. Диалоговое окно файла закрывается, в поле «Файл» появляется полное имя файла</p> <p>5. Выбранный файл появляется в списке файлов галереи</p>
--------------	---	-----	---------	----------------	---	--

Идентификатор

Модуль и подмодуль

Исходные данные, необходимые для выполнения теста

Шаги



Свойства хорошего тест-кейса

Хороший тест-кейс удовлетворяет следующим критериям:

- Обладает **высокой вероятностью обнаружения ошибки.**
- Исследует **соответствующую («ту, которую надо») область приложения.**
- Выполняет какие-то **интересные действия.**
- **Не выполняет ненужных действий.**
- Является **не слишком простым, но и не слишком сложным.**
- **Не является избыточным** по отношению к другим тестам.
- Делает обнаруженную ошибку **очевидной.**
- Позволяет **легко диагностировать** ошибку.

Вопросы



И сейчас «поиграем» 😊

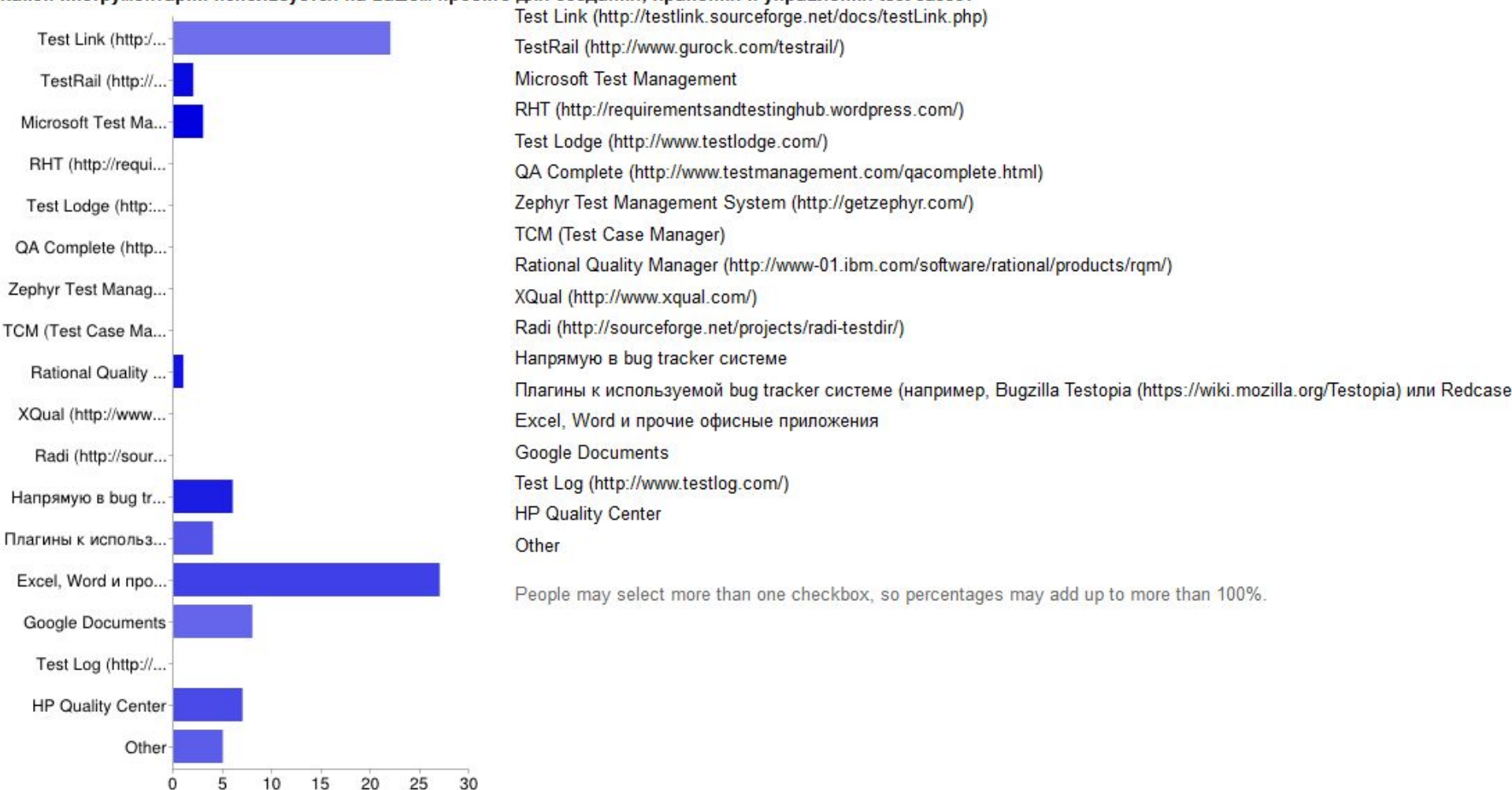
Тестовый сценарий (test scenario) – набор тестов (тест-кейсов), собранных в последовательность для достижения некоторой цели.

Хороший тестовый сценарий всегда **следует некоторой логике**, например: типичному использованию приложения, удобству тестирования, распределению функций по модулям и т.д.



Какой инструментарий используется на вашем проекте для создания, хранения и управления test cases?

Какой инструментарий используется на вашем проекте для создания, хранения и управления test cases?



People may select more than one checkbox, so percentages may add up to more than 100%.

Тестовые сценарии: рекомендации

Microsoft Excel - Example_TC

File Edit View Insert Format Tools Data Window Help Adobe PDF

B1

	A	B	C	D	E	F	G	H	I
1	UI_001	(All) (Top 10...) (Custom...) A B C	R1001	Gallery	Upload	Load File, Correct 1. Open load dialog 2. Choose file 3. Click OK 4. Close window			
2	UI_001		R1001	Gallery	Upload	Load File, Incorrect 1. Open load dialog 2. Choose file 3. Click OK 4. Close window			
3	UI_001.3	A	R1001	Gallery	Upload	Load File, Correct 1. Open load dialog 2. Choose file 3. Click OK 4. Close window			
4	UI_001.4	B	R1001	Gallery	Upload				
5	UI_001.5	A	R1001	Gallery	Upload				
6									
7									
8									
9									
10									
11									
12									
13									

Используйте фильтры

Используйте группировку

Используйте отдельные листы

Smoke test / Critical path test / Sheet3 /

Filter Mode

Шаги разработки тестов



1. Начинайте **как можно раньше**, ещё до выхода первого билда.



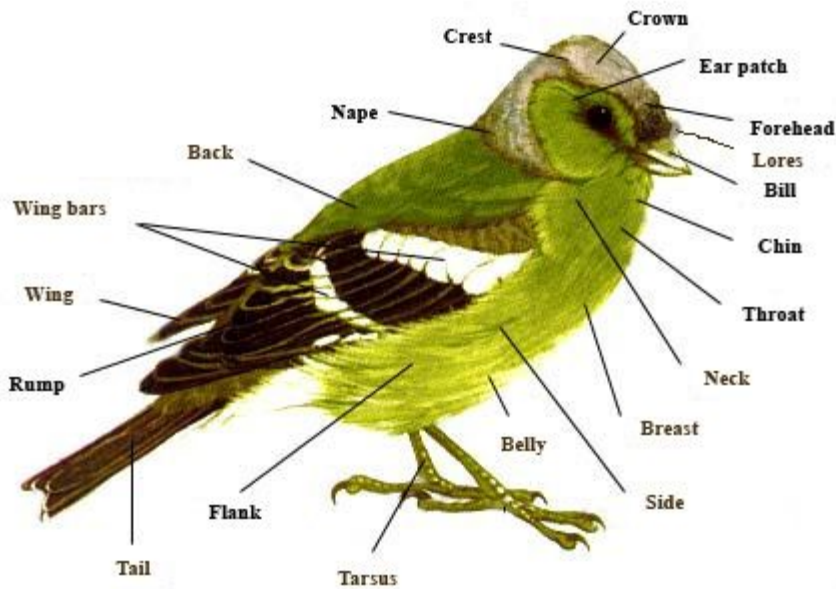
2. Разбивайте
приложение на
отдельные
части/модули.



3. Для каждой
области/модуля
пишите чек-лист.



4. Пишите **вопросы**,
уточняйте детали,
добавляйте
«косметику»,
используйте **copu-paste**.



5. Получите **рецензию**
коллег-тестировщиков,
разработчиков,
заказчиков.



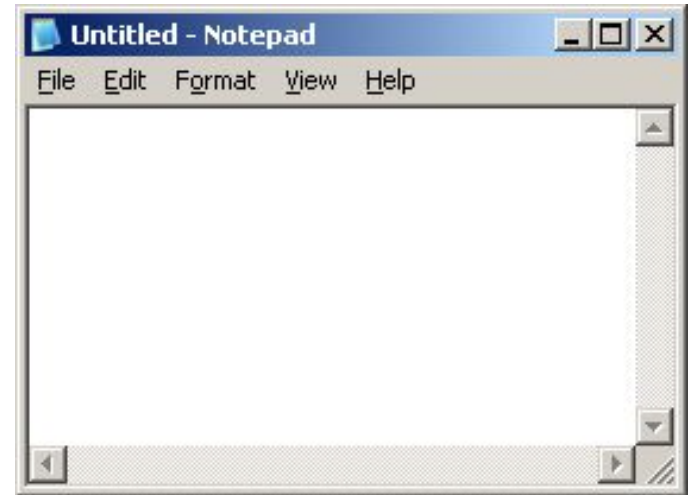
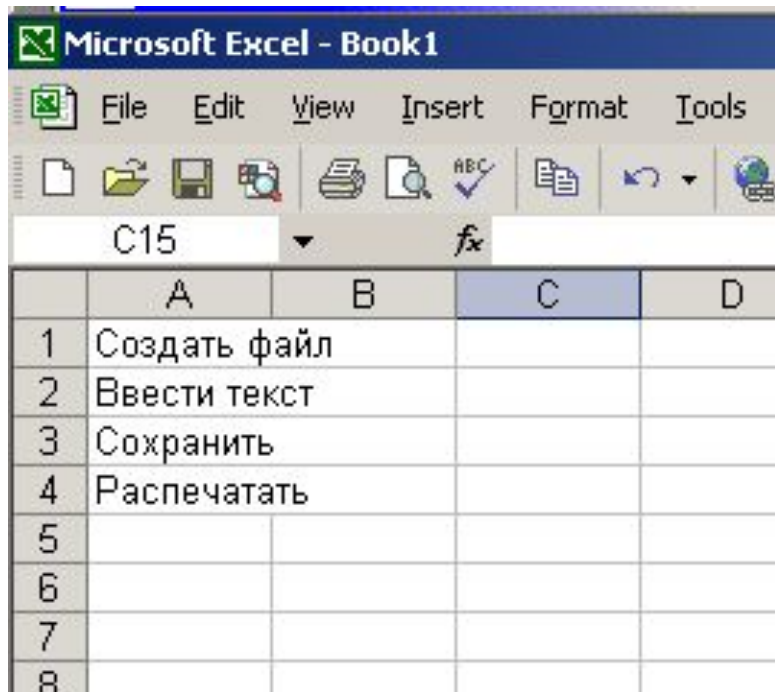
6. Обновляйте тесты,
как только обнаружили
ошибку или изменилась
функциональность.



Пример разработки тестов

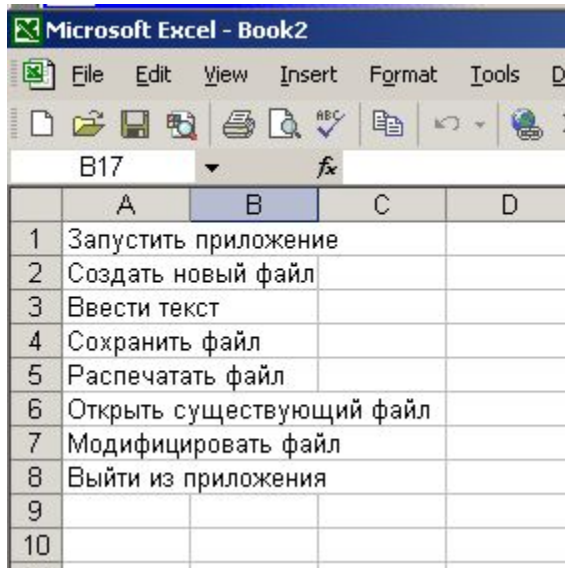


1. Что такое Notepad?
2. Какие функции для него наиболее важны?



Что ещё?

Итак, вот наш Smoke test



The screenshot shows a Microsoft Excel window titled "Microsoft Excel - Book2". The menu bar includes File, Edit, View, Insert, Format, and Tools. The toolbar contains icons for file operations and editing. The active cell is B17. The following table is displayed in the spreadsheet:

	A	B	C	D
1	Запустить приложение			
2	Создать новый файл			
3	Ввести текст			
4	Сохранить файл			
5	Распечатать файл			
6	Открыть существующий файл			
7	Модифицировать файл			
8	Выйти из приложения			
9				
10				

Перенесём его в шаблон для разработки тестов.

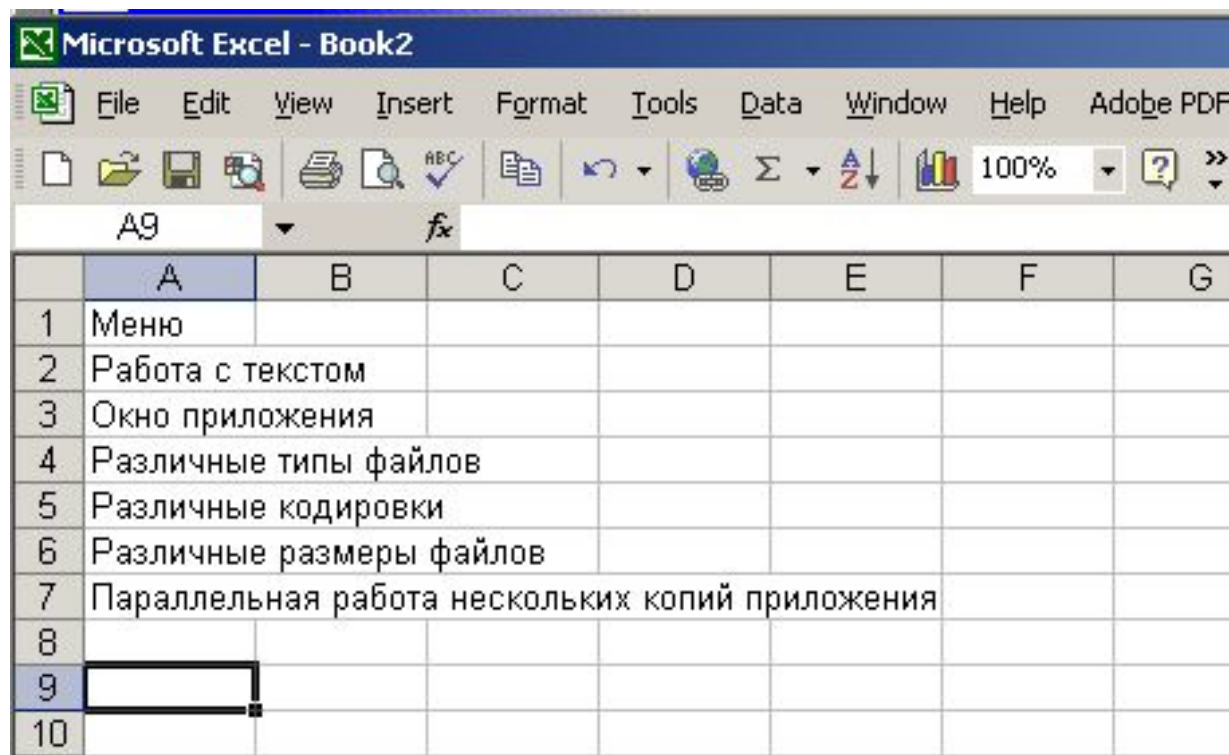
Фактически, это – чек-лист. И сами пункты грамотно сформированного чек-листа – готовые заголовки тест-кейсов.

"Notepad"					
Функциональные тест-кейсы					
Идентификатор	Ссылка на требование	Модуль	Подмодуль/ экран	Описание теста	Ожидаемый результат
Тестировал(и):					
Дата(даты) тестирования:			ОС:		
				Запустить приложение	
				Создать новый файл	
				Ввести текст	
				Сохранить файл	
				Распечатать файл	
				Открыть существующий файл	
				Модифицировать файл	
				Выйти из приложения	

Когда мы распишем наши тесты по правилам, Smoke Test примет следующий вид:

	A	B	C	D	E	F	G
4	Идентификатор	Ссылка на требование	Модуль	Подмодуль/ экран	Описание теста	Ожидаемый результат	Статус ("не тестировано", "выполнено успешно", "выполнение завершилось ошибкой")
5	ST_001	R1	Приложение не запущено		Запустить приложение 1. Выполнить команду notepad из командной строки	1. Появляется окно notepad с пустым файлом	Не тестировано
6	ST_002	R1, R16	Приложение		Создать новый файл 1. Выполнить последовательность команд "Файл" -> "Создать" с использованием меню	1. Создаётся новый файл (в рабочей области приложения пусто)	Не тестировано
7	ST_003	R34, R75.7	Приложение		Вести текст 1. Набрать несколько слов 2. Удалить несколько слов	1. В рабочей области приложения отображается набранный текст 2. Удаляемые слова пропадают из рабочей области приложения	Не тестировано
8	ST_004	R23	Приложение	Работа с файлами	Сохранить файл 1. Создать новый файл. Ввести немного текста. 2. Выполнить последовательность команд "Файл" -> "Сохранить" с использованием меню 3. Выбрать каталог для сохранения файла и ввести имя файла 4. Нажать кнопку "Сохранить"	1. Создаётся новый файл, введённый текст отображается в рабочей области приложения 2. Появляется диалоговое окно "Сохранить файл" Какой каталог для сохранения должен отображаться по умолчанию? 3. Имя файла отображается в строке ввода 4. Диалоговое окно "Сохранить файл" исчезает, на диске в указанном каталоге появляется сохранённый файл	Не тестировано
9	ST_005	R45, R57, R92	Приложение	Работа с файлами	Распечатать файл 1. Выполнить последовательность команд "Файл" -> "Печать" с использованием меню 2. Следовать инструкциям	1. Открывается диалог "Печать документа" Какова реакция приложения на отсутствие в системе установленных принтеров?	Не тестировано

Аналогичным образом начинаем и продолжаем работать с тестом критического пути:



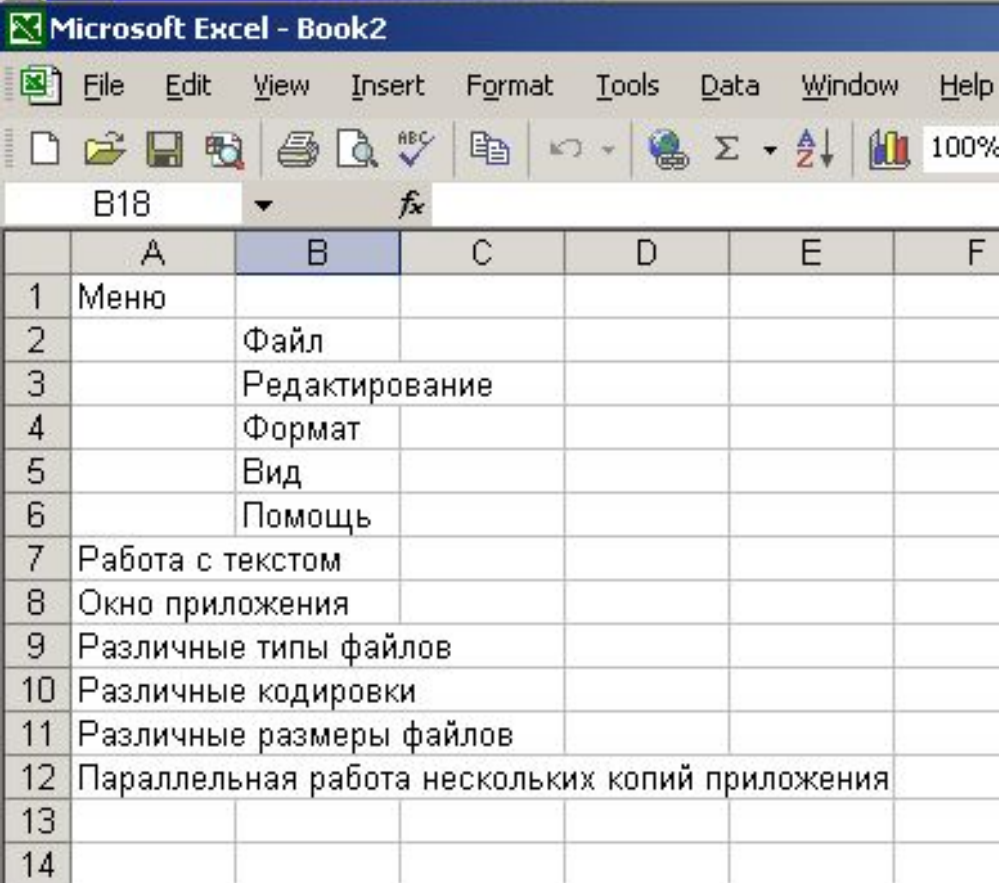
Microsoft Excel - Book2

File Edit View Insert Format Tools Data Window Help Adobe PDF

A9 fx

	A	B	C	D	E	F	G
1	Меню						
2	Работа с текстом						
3	Окно приложения						
4	Различные типы файлов						
5	Различные кодировки						
6	Различные размеры файлов						
7	Параллельная работа нескольких копий приложения						
8							
9							
10							

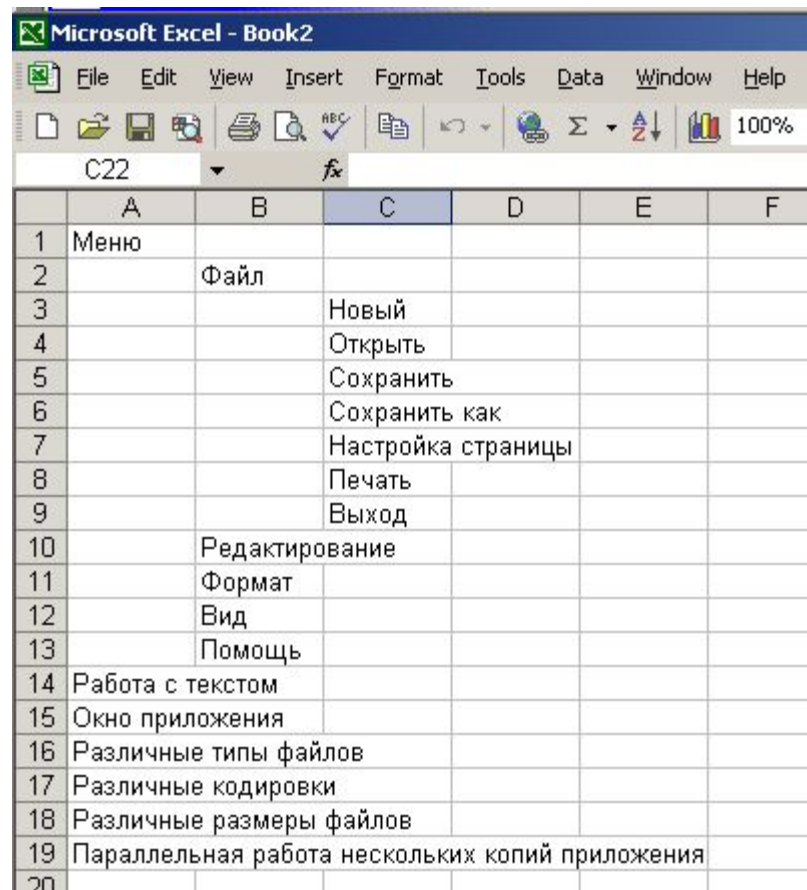
Детализируем чек-лист:



The screenshot shows the Microsoft Excel interface with a spreadsheet containing a checklist. The menu bar includes File, Edit, View, Insert, Format, Tools, Data, Window, and Help. The toolbar contains various icons for file operations and editing. The active cell is B18. The spreadsheet data is as follows:

	A	B	C	D	E	F
1	Меню					
2		Файл				
3		Редактирование				
4		Формат				
5		Вид				
6		Помощь				
7	Работа с текстом					
8	Окно приложения					
9	Различные типы файлов					
10	Различные кодировки					
11	Различные размеры файлов					
12	Параллельная работа нескольких копий приложения					
13						
14						

Продолжаем детализацию до тех пор, пока не получим логичный и достаточный набор тестов. После этого переносим его в шаблон и работаем аналогично тому, как мы делали это при разработке Smoke Test.



The screenshot shows a Microsoft Excel spreadsheet with a menu structure. The menu items are organized as follows:

	A	B	C	D	E	F
1	Меню					
2		Файл				
3			Новый			
4			Открыть			
5			Сохранить			
6			Сохранить как			
7			Настройка страницы			
8			Печать			
9			Выход			
10		Редактирование				
11		Формат				
12		Вид				
13		Помощь				
14	Работа с текстом					
15	Окно приложения					
16	Различные типы файлов					
17	Различные кодировки					
18	Различные размеры файлов					
19	Параллельная работа нескольких копий приложения					
20						

Есть вопросы? Давайте обсудим!



Как делать не нужно

Ввод данных:

1. Ввести в окно цифры
2. Проверить простейшие тригонометрические операции (1*1=1)
3. Проверить деление числа на ноль
4. Проверить наибольшие и наименьшие значения с которыми калькулятор может работать
5. Проверка ничего не вводить в поле для ввода

1. Вводятся данные
2. Появляется конечный результат (В результате получилось: 1)
3. Появляется ошибка "делить на ноль нельзя"
4. Вводится диапазон от 1 до 32
- 2.5. Появляется ошибка "не является допустимым значением с плавающей точкой "

Как делать хорошо

Проверка всех возможных вариантов ввода данных

- | | |
|---|---|
| 1. Запустите Калькулятор <code>c:\calc.exe</code> | 1. Приложение запустилось.
В ПВВР отображается 0 |
| 2. В GUI нажмите мышью 10,23
±456789 | 2. В ПВВР отобразилось -10,23456789 |
| 3. В GUI нажмите кнопку “C” | 3. В ПВВР отображается 0 |
| 4. С верхнего ряда цифр. Кл-ры
введите 10,23456789 | 4. В ПВВР отобразилось 10,23456789 |
| 5. Нажмите кнопку “C” | 5. В ПВВР отображается 0 |
| 6. С доп. цифр. Кл-ры введите
10,23456789 | 6. В ПВВР отобразилось 10,23456789 |
| 7. Нажмите в GUI кнопку “<” | 7. В ПВВР отображается 10,2345678 |
| 8. Нажмите “backspace” | 8. В ПВВР отображается 10,234567 |
| 9. Нажмите кнопку “C” | 9. В ПВВР отображается 0 |
| 10. Запустите Блокнот, наберите в нём
символьную последовательность
-10,23456789, скопируйте её в буфер
обмена, перейдите в приложение
Калькулятор и нажмите Ctrl+V | 10. В ПВВР отобразилось -10,23456789 |