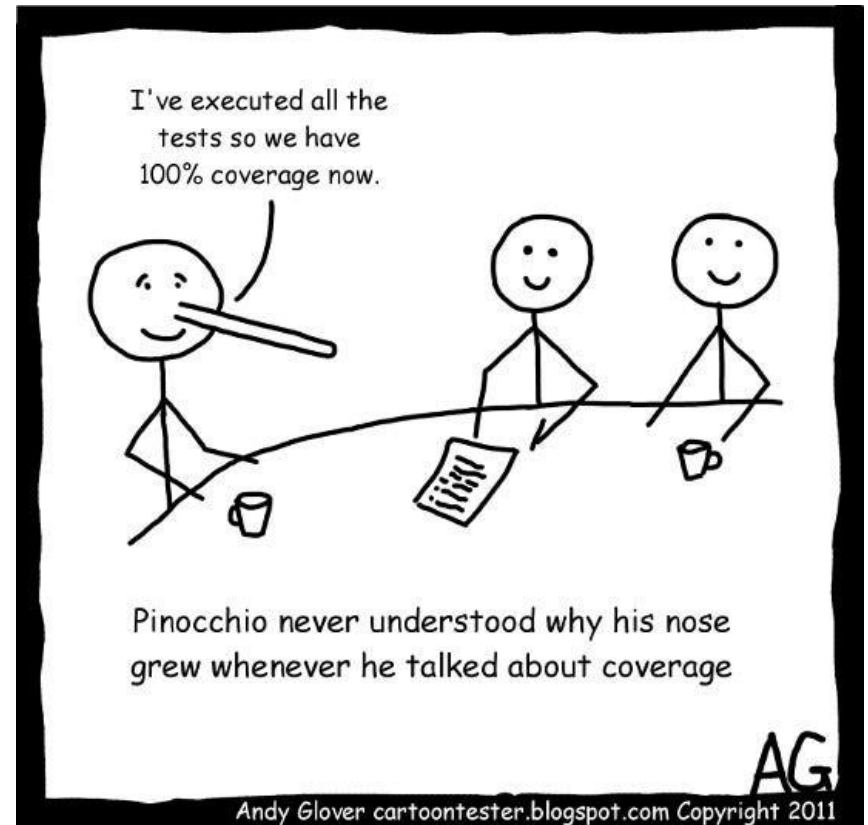


# QA Theory. Лекция 5

- Создание тестов
  - Тест анализ
  - Тест дизайн



## Основы QA процесса

### SDLC

Анализ

Дизайн

Кодирование

Тестирование

Внедрение

### STLC

Планирование

Тест анализ  
*Что тестируем?*

Тест дизайн  
Разработка тестов  
*Как тестируем?*

Выполнение тестов

Анализ и отчет о  
результатах  
тестирования

**Мониторинг и контроль –**  
непрерывное сравнение актуального прогресса с планом тестирования

## Тест анализ

Отвечаем на вопрос: **Что мы тестируем?** Необходимо глубоко изучить продукт – функциональные требования, диаграммы, схемы и т.д.

Проверяем документацию на отсутствие противоречий

Составляем **Test scenarios** (High level test conditions) : В одну строку прописываем - что нам надо проверить

Это удобно для понимания бизнес пользователей

В рамках анализа тестирования, нам нужно определить следующие вещи:

- Уровень детализации и качества наших требований
- Связи между требованиями
- Уровни тестирования — на сколько глубоко нам нужно протестировать каждое требование (требуется ли интеграционное, системное тестирование, и т.д.)
- Тестовая среда / уровень подготовки команды
- Инструменты по управлению тестированием (Test management system)
- Доступность для консультации других участников проекта

Цель:

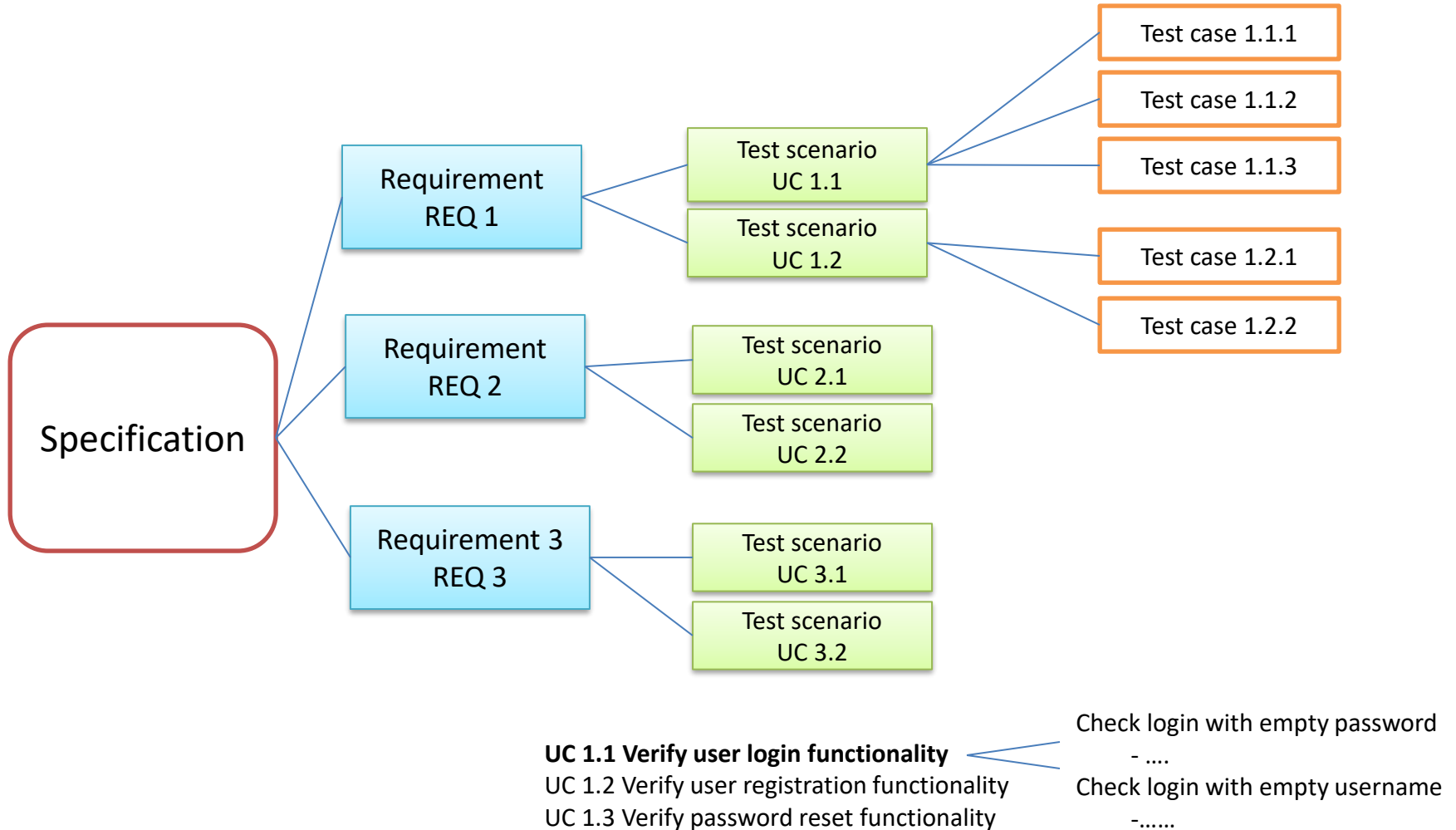
- Создание ясности благодаря определению точных целей тестов. Мы знаем что именно и на сколько глубоко нужно тестировать.
- Предотвращение дефектов: именно на этапе анализа тестирования проводится [статическое тестирование](#).

# Тест дизайн

Отвечаем на вопрос – **Как будем тестировать?**

- На основе **Test scenario** создаем **Test case**  
**Test case** – пошаговая инструкция проверки определенного test scenario
- Объединяем test case в Test suits (тестовые наборы). Группируем для удобства:
  - тесты для одной страницы
  - тесты для определенного функционала
  - тесты по платформам (Chrome, Firefox)
  - По назначению – тестирования удобства пользования, тестирование безопасности, ...
- Устанавливаем приоритеты
- Подготавливаем тестовые данные
- Формируем требования и настраиваем тестовую среду, инструменты. Например, для мобильного тестирования нужны разные устройства с разной версией
- Traceability matrix
  - Test scenario ----Test case
  - Business requirements ---Test scenarios ---- Test cases ---Defects

## Тестовая модель



<https://www.browserstack.com/guide/how-to-create-test-scenarios>

## Test Scenarios

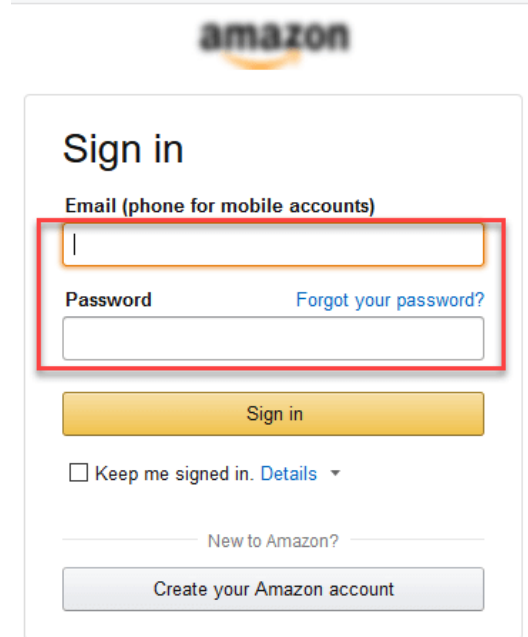
**Test Scenario 1:** Check the Search Functionality

**Test Scenario 2:** Check the Payments Functionality

**Test Scenario 3:** Check the Login Functionality

### Test cases for the Test Scenario 3: “Check the Login Functionality”:

1. Check system behavior when valid email id and password is entered.
2. Check system behavior when invalid email id and valid password is entered.
3. Check system behavior when valid email id and invalid password is entered.
4. Check system behavior when invalid email id and invalid password is entered.
5. Check system behavior when email id and password are left blank and Sign in entered.
6. Check Forgot your password is working as expected
7. Check system behavior when valid/invalid phone number and password is entered.
8. Check system behavior when “Keep me signed” is checked



The image shows the Amazon sign-in page. At the top is the Amazon logo. Below it is the heading "Sign in". There are two input fields: "Email (phone for mobile accounts)" and "Password". The "Email" field is highlighted with a red border. To the right of the "Password" field is a link that says "Forgot your password?". Below the input fields is a yellow "Sign in" button. Underneath the button is a checkbox labeled "Keep me signed in." with a link "Details" to its right. At the bottom, there is a link "New to Amazon?" and a button "Create your Amazon account".

## Functionality matrix. You can't manage what you can't measure

Id	Functionality
1.	Calculator view
1.1.	Standard
1.1.1.	"+" function
1.1.2.	"-" function
1.1.3.	"%" function
...	
1.2.	Scientific
1.2.1.	"+" function
1.2.2.	"-" function
1.2.3.	"exp" function
1.2.4.	"mod" function
...	
1.3.	Graphing
...	
2.	Converter view
1.1.	Currency
...	
3.	About
3.1	
...	

## Traceability matrix

[illegible]



## Состав Test case (основные поля)

Filed	Поле	Example
Test case ID	Идентификатор	TC-1.1.1
Test case description	Описание	Check login with empty password
Prerequisites	Предварительные условия	A valid user account Login to URL: https://test.qatest.com
Test steps	Необходимые действия	1. Enter username (data: <b>user342</b> ) 2. Enter password (data: <b>empty</b> ) 3. Click «Log in»
Expected result	Ожидаемый результат	Login failed
Actual result	Полученный результат	
Status	Статус/Отметка о прохождении	Not executed/Passed/Failed/...

### Что нужно стараться избегать:

- Зависимость от других тестов (*для выполнения этого теста надо сначала сделать другой*)
- Нечеткой формулировки (*тест должен быть понятен другим*)
- Отсутствия необходимой для прохождения информации, Излишней детализации

### Pass / Fail criteria. Что будет Pass criteria для:

- Создания объекта
- Удаления объекта
- Изменения
- Бизнес операции

# Test case example

Test Case ID	BU_001	Test Case Description	Test the Login Functionality in Banking		
Created By	Mark	Reviewed By	Bill	Version	2.1

QA Tester's Log	Review comments from Bill incorporate in version 2.1
-----------------	--

Tester's Name	Mark	Date Tested	1-Jan-2017	Test Case (Pass/Fail/Not)	Pass
---------------	------	-------------	------------	---------------------------	------

S #	Prerequisites:
1	Access to Chrome Browser
2	
3	
4	

S #	Test Data
1	Userid = mg12345
2	Pass = df12@434c
3	
4	

Test Scenario	Verify on entering valid userid and password, the customer can login
---------------	--

Step #	Step Details	Expected Results	Actual Results	Pass / Fail / Not executed / Suspended
1	Navigate to http://demo.guru99.com	Site should open	As Expected	Pass
2	Enter Userid & Password	Credential can be entered	As Expected	Pass
3	Click Submit	Cutomer is logged in	As Expected	Pass
4				

+

- Priority
- Run type (manual)
- Test case status: 1 –Design, 2-Ready for review, 3-Rework required, 4-Approved, 5-Obsolete, 6-N/A
- Regression flag: Y/N
- Eligible to Automate: Y/N

## Техники тест дизайна (для black-box testing)

1. **Equivalence partitioning** – Тестирование на основе классов эквивалентности. Деление входных данных на группы эквивалентности по своему влиянию на систему. Включаем как позитивные так и негативные значения
2. **Boundary value testing** – Техника анализа граничных значений. Проверка поведения системы на граничных входных данных. Подвид equivalence partitioning для цифровых значений
3. **State - Transition testing** – Тестирование на основе состояний переходов. Применяется для фиксирования требований и описания дизайна приложения
4. **Decision Table Testing** – техника основанная на методе черного ящика, которая применяется для систем со сложной логикой
5. **Pair-wise testing** – Метод пар

## Equivalence partitioning

Протестировать поля ввода — **Фамилия**

Условие: в поля ввода можно внести

- русские/английские буквы
- заглавные и строчные буквы
- “-” (дефис)

Тест кейс	Класс эквивалентности	Тестовые значения
<b>Допустимые значения (блок позитивных тестов):</b>		
1	Все буквы русского (от А до Я)/английского (A-Z)	ПФ
2	Заглавные и строчные буквы	РgЯя
3	Фамилии через дефис	Салтыков-Щедрин
<b>Недопустимые значения (блок негативных тестов):</b>		
4	Цифры (от $-\infty$ до $+\infty$ )	1026
5	Спец. символы, отличные от дефиса	%\$#/

Протестировать расчет скидки в зависимости от кол-ва товаров

Условия:

- До 5 товаров: 0%
- 5-10 товаров: 5%
- 10-15 товаров: 15%
- Больше 15 товаров: 20%

Тест кейс	Класс эквивалентности	Тестовое значение
1	1-4 товаров	3
2	5-9 товаров	7
3	10-14 товаров	12
4	> 15 товаров	21

## Boundary value testing

Протестировать расчет скидки в зависимости от кол-ва товаров. Условия:

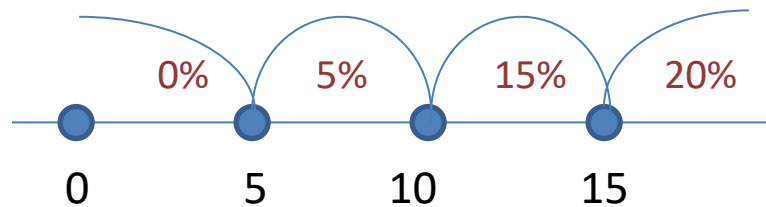
- До 5 товаров: 0%
- От 5 до 10 товаров: 5%
- От 10 до 15 товаров: 15%
- Больше 15 товаров: 20%

Тестовые данные для граничных значений

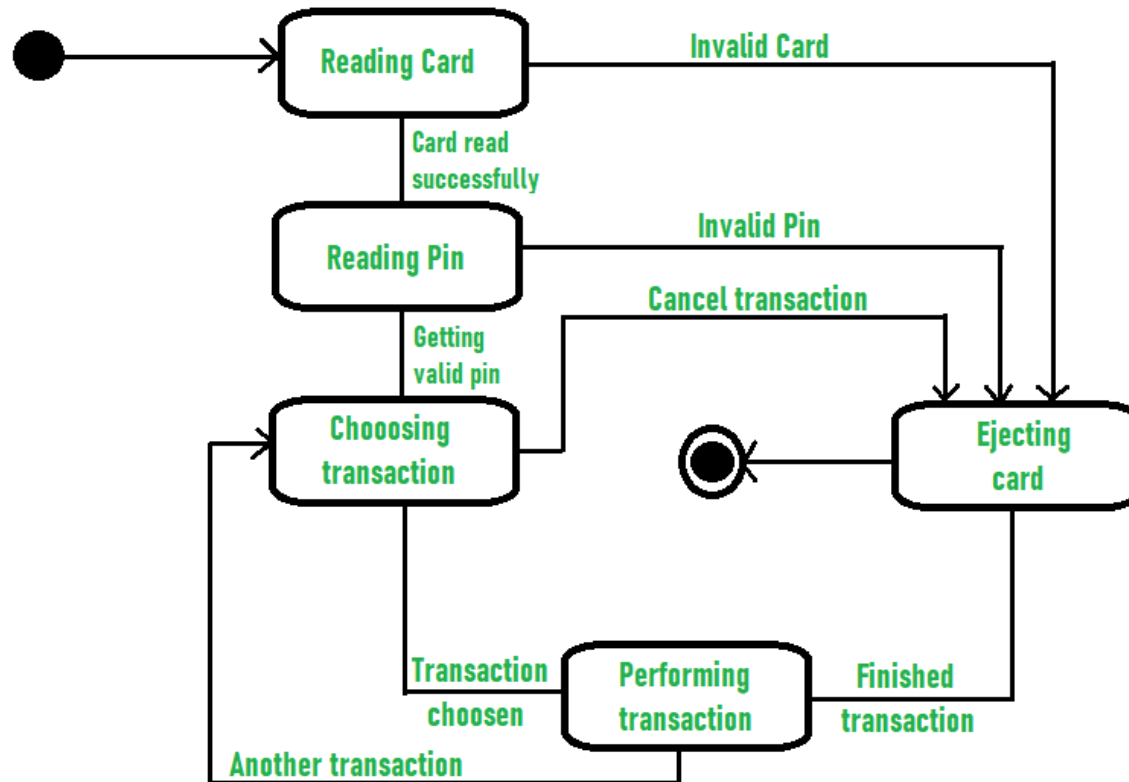
-1, 0, 1
4, 5, 6
9, 10, 11
14, 15, 16

Тестовые данные для классов эквивалентности

3
7
12
21



## State - Transition testing



**State Transition Diagram for ATM System**

## Decision Table Testing


- Расчет скидки на страхование жизни в зависимости от
  - Семейного положения
  - Пола
  - Образования


Условие	T1	T2	T3	T4			Tn
В браке?	Да	Нет	Да	Нет			Нет
Пол	М	М	Ж	Ж			М
Образование	В	В	В	В			С
Результат	10%	8%	15%	9%			0%


	Test1	Test2	Test3	Test4
Login	Valid	Valid	Empty	Empty
Password	Valid	Empty	Valid	Empty
Authorization	Passed	Failed	Failed	Failed

## Pair-wise testing

**Search for a Vehicle**

Year 

Make 

Model 

Make	Model	Color
2021	Toyota	Black
2022	Audi	White
		Red

	Make	Model	Color
1	2021	Toyota	Black
2	2021	Toyota	White
3	2021	Toyota	Red
4	2022	Toyota	Black
5	2022	Toyota	White
6	2022	Toyota	Red
7	2021	Audi	Black
8	2021	Audi	White
9	2021	Audi	Red
10	2022	Audi	Black
11	2022	Audi	White
12	2022	Audi	Red



<https://www.pairwise.org/tools.html>

	Color	Model	Color
1	Red	Toyota	2021
2	Red	Audi	2022
3	White	Audi	2021
4	White	Toyota	2022
5	Black	Toyota	2022
6	Black	Audi	2021

<https://pairwise.teremokgames.com/>



## **Домашнее задание**

<https://playground.learnqa.ru/puzzle/triangle>