

Типове и нива на тестване

Юнит тестване / Интеграционно тестване /
Системно тестване / Приемно тестване
Функционално и нефункционално тестване



СофтУни

Преподавателски екип



SoftUni



Софтуерен университет

<http://softuni.bg>

sli.do

#QA-Basics

1. Нива на тестване

- Юнит тестване (компонентно тестване)
- Интеграционно тестване
- Системно тестване
- Тестване за приемане от краен потребител (приемно тестване)

2. Типове тестване

- Функционално тестване
- Нефункционално тестване





Нива на тестване

Юнит тестване / Интеграционно тестване /
Системно тестване / Приемно тестване

Нива на тестване

- Групи от **тестови дейности**
- Всяко **ниво** е **част** от **тестовия процес**
- **Съответства** на **определена фаза** от разработката на софтуера
- **Тест нива:**
 - Тестване за **одобрение** / **Приемно** тестване (Acceptance testing)
 - **Системно** тестване (System testing)
 - **Интеграционно** тестване (Integration testing)
 - **Юнит** тестване / **Компонентно** тестване (Unit testing)



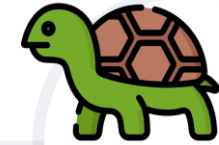
Йерархия на тестовите нива

Приемно тестване
(Acceptance testing)

Системно тестване
(System testing)

Интеграционно тестване
(Integration testing)

Юнит тестване
(Unit testing)

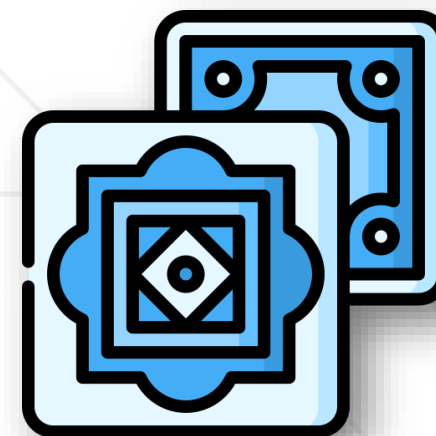
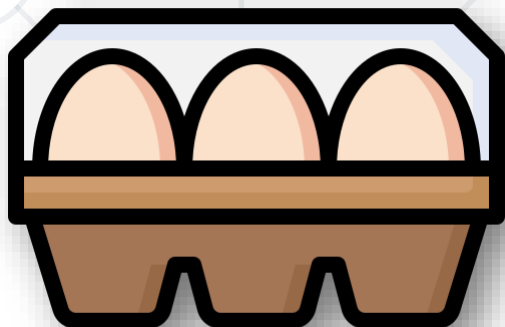




Юнит тестване

Тестване на отделен компонент

- Проверяване на всяко яйце от кутията, преди закупуването ѝ



- Проверяване дали фаянсовата плочка не е счупена, преди нанасянето на лепило

- Какво е юнит тестване / компонентно тестване?
 - **Първото** или **най-базово ниво** на тестване
 - Тества **отделни компоненти** на софтуера
 - Компонент може да бъде отделна функция, метод, модул или обект
 - Обикновено се **извършва** от **самите програмисти** във фазата на писане на **код**
 - Изпълнява се изолирано
- **Защо ни е необходимо?**
 - Проверява дали отделните **компоненти работят коректно**
 - **Позволява** дефектите да бъдат отстранени **рано**, още във фазата на разработка

- Проверка на възрастта

Програмата задава въпрос: "На колко години си?"
Прочита вевъдените от потребителя "години"
ако годините са ≥ 18
връща вярно
в противен случай
връща невярно

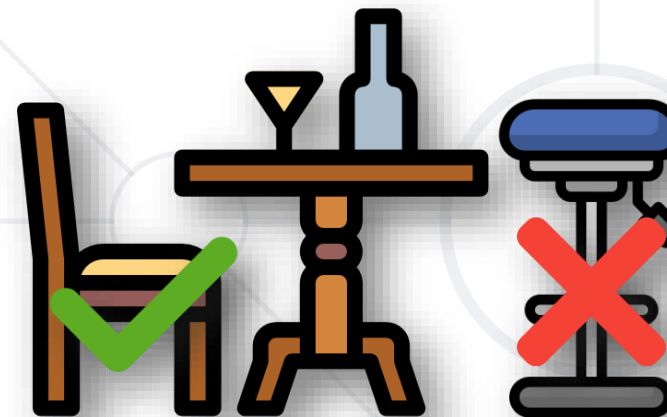
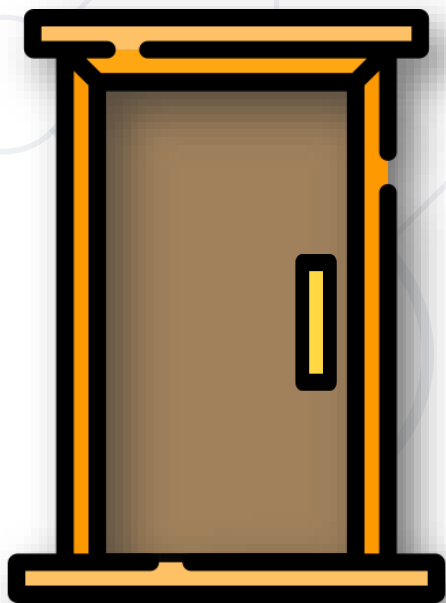
Положителен тест(20) → очаква се вярно
Отрицателен тест (16) → очаква се невярно
Граничен тест (18) → очаква се вярно



Интеграционно тестване

Тестване на взаимодействието между компоненти

- Врата и каса за врата



- Столове подходящи за маса

- Какво е интеграционно тестване?
 - **Второ ниво** от процеса по тестване на софтуер
 - Отделните компоненти или единици на софтуера се **тестват в група**
 - Извършва се от програмисти, QA специалисти или специални интеграционни екипи
 - Предполага се, че компонентите **вече са тествани поотделно**



- **Защо ни е необходимо интеграционното тестване?**
 - След свързването на отделните компоненти един с друг, може да възникне нова грешка
 - Тестването трябва да потвърди, че **всички свързани компоненти си взаимодействат правилно**
 - Основната цел е да се открият грешките в:
 - **Интерфейси**
 - **Взаимодействието** между **интегрирани** компоненти
 - **Взаимодействието** между **системи**

- **Вътрешно** интеграционно тестване
 - Разкрива дефекти в интерфейсите и взаимодействието между интегрираните компоненти
 - "Integration test **in the small**"
- **Външно** интеграционно тестване
 - Тестване на съчетанието на системи и пакети
 - Тестване на интерфейси към външни организации
 - "Integration test **in the large**"

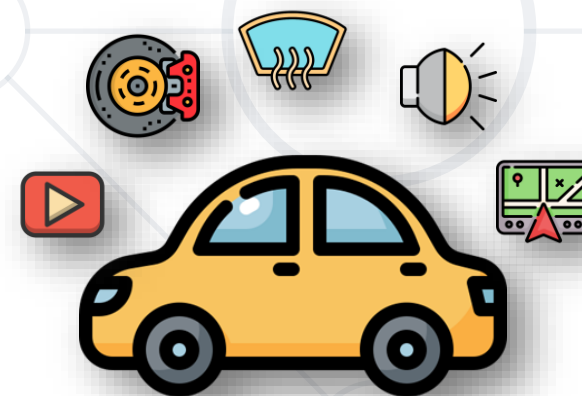
- **GitHub** има няколко модула (компоненти):
 - **Home Page → Login Page → User Dashboard**
- Всеки от тях е тестван поотделно
- Искаме да проверим дали **работят заедно**
- **Интеграционни тестове:**
 - Тестваме дали бутонът за вход води към формата за вход
 - Тестваме дали след успешно влизане с потребителско име и парола, се показва потребителският дашборд
 - Тестваме дали след излизане от профила, потребителският дашборд е недостъпен



Системно тестване

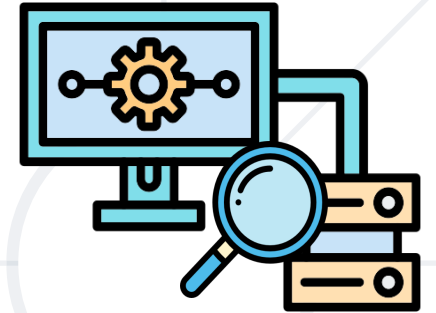
Тестване на цялата система

- В автомобилната индустрия, всеки произведен автомобил е щателно тестван, в края на производствения процес:
 - Двигател, джанти, волан, спирачки
 - Врати, ключалки, седалки
 - Електрическа система: светлини, чистачки, климатик
 - Мултимедийна система: радио, GPS, карти
 - и т.н.
- QA-те вече са оценили всички тези функционалности поотделно, а също и взаимодействащи една с друга, но те трябва да бъдат **тествани и като цялостна система**



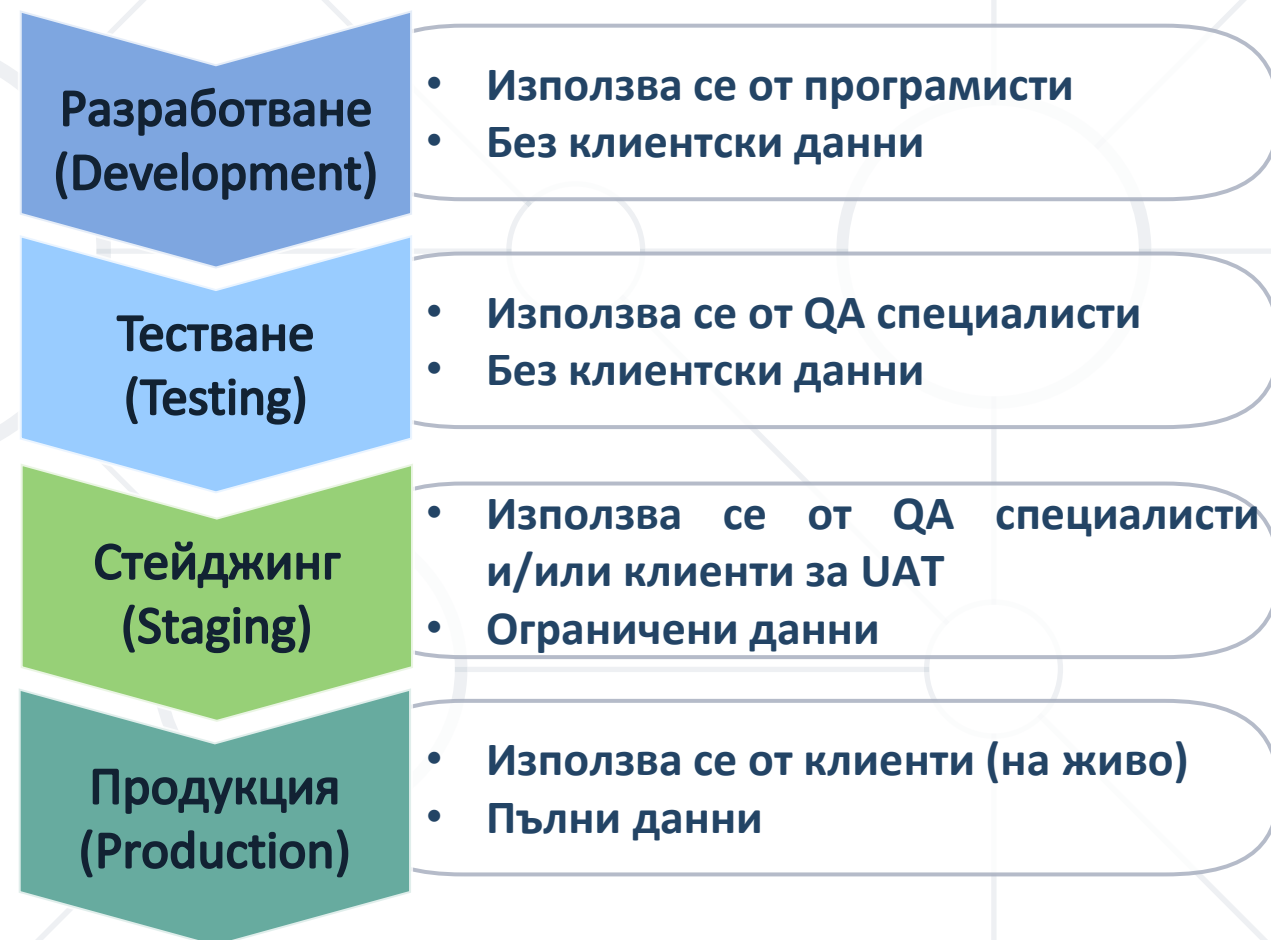
- Какво е системно тестване?

- **Трето ниво** от процеса по тестване на софтуер
- С фокус върху **цялата система**:
 - Нейното **поведение** (какво прави системата)
 - Колата правилно ли е сглобена, работи ли по предназначение?
 - Нейните **възможности** (как се справя системата)
 - Дали автомобилът е надежден, сигурен, в добро състояние, каква е неговата производителност и ефективност
- Реализира се чрез тестване **"от край до край"** (E2E, end-to-end)
- Извършва се само от **QA специалисти**



- **Защо ни е необходимо системно тестване?**
 - **Предишни тестове** са били изпълнени спрямо **технически спецификации**
 - **Системните тестове** разглеждат системата от гледна точка на крайния потребител
 - Системните тестове проверяват дали всички компоненти на дадена система функционират при реален сценарии
 - Системното тестване може да бъде **функционално** и **нефункционално**

- Системното тестване изисква специално обособена **"стейджинг" среда**
- **Максимално точно копие** на сайта/приложението, до което имат достъп крайните потребители, предназначено за **системно тестване**



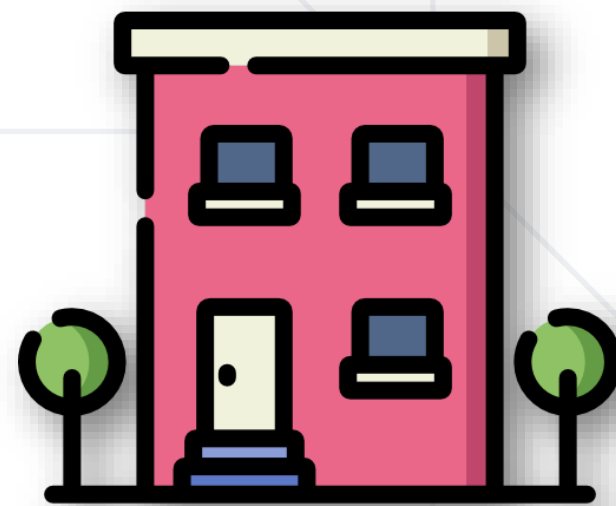
**UAT - User Acceptance Testing /
Потребителско тестване за одобрение*



Тестване за приемане от крайния клиент

Клиентът тества крайния продукт от бизнес
гледна точка

- Собственик на апартамент, **проверява апартамента след ремонт**
 - Всички стаи: подове, тавани, стени, врати, прозорци
 - Уреди
 - Мебели
 - Вода и водопровод
 - Електричество
 - Газ
 - И Т.Н.



- Какво е тестване за приемане от крайния клиент?
 - **Последното ниво**, обикновено преди внедряване (deployment)
 - Валидира **цялостно функционално бизнес решение**
 - Под внимание се вземат **законовите** и/или **регулаторните** изисквания
 - Приемното тестване се изпълнява:
 - От членове на **бизнес** екипа (**алфа** тестване)
 - От **крайни** потребители (**бета** тестване)
 - Следват се **оперативни** инструкции
 - Гарантира се спазване на **договорните** и **регулаторни** насоки

- **Защо ни е необходимо тестване за приемане от крайния потребител?**
 - Проверява **работата** на **системата**, обикновено преди внедряване
 - Основната **цел** е **работещо бизнес решение**
 - Не се фокусира върху козметичните грешки
 - Отговаря на въпроса, дали **актуалното поведение на системата съответства** на очакванията на клиента

- Най-новият Microsoft Windows се тества първо локално в Редмънд (алфа тестване), след това от външни потребители (навсякъде по света)

Алфа тестери

- Група **вътрешни** потребители
- **Запознати** са с проекта
- **Не участват** пряко в развитието му
- Тестват дали приложението **работи правилно**
- **Дават обратна информация** за това как **потребителското изживяване** може да се подобри

Бета тестери

- След алфа тестването, грешките се коригират
- **Избрана група крайни потребители**
- Служи като "**плавен старт**"
- **Обратна връзка** от **реални потребители**, които **нямат предварителни познания** за приложението



Типове тестване

Функционално и нефункционално тестване

- Група от **тест дейности**, които **тестват специфични характеристики** на определена софтуерна система
- Типовете тестове се разделят на **две** основни групи:
 - **Функционално** тестване
 - Отговаря на въпроса "**Какво?**"
 - Потвърждава **правилното функциониране** на софтуера
 - **Нефункционално** тестване
 - Отговаря на въпроса "**Как?**"
 - Потвърждава **ефективността** и **производителността** на софтуера

- **Функционалното тестване** на софтуер за онлайн банкиране включва:
 - Тестване дали средствата са **точно преведени**
 - Дали лихвените **изчисления са правилни**
 - Дали плащанията по сметки се **извършват навреме**
- **Нефункционално тестване** се фокусира върху **сигурността на системата**:
 - Да се гарантира, че достъпът е напълно **безопасен**
 - Да се гарантира, че системата може да се **справи с натоварването**
 - Особено в пикови периоди, като началото на месеца, когато масово се изплащат заплати и други плащания към бюджета

Типове тестване и нива на тестване

- **Типовете тестване** могат да се **прилагат** на много / всички **тестови нива**
- Пример: тестване на сценарий "регистрай потребител"
 - **Функционални** тестове:
 - Валидна потребителска информация, невалидна потребителска информация, дублирана потребителска информация
 - **Нефункционални** тестове:
 - Производителност (100 хиляди потребители), надеждност (по 1 потребител в секунда за 24 часа), UX тест (навигацията лесна ли е за потребителя)





Функционално тестване

Тестване на определени функции

- **Сешоар: Функционално тестване**

- Старт / Стоп
- Промяна на силата
- Горещ / студен въздух
- Йонни настройки
- Тестове на различни приставки
- Тестове на прибиращия се кабел
- И Т.Н.



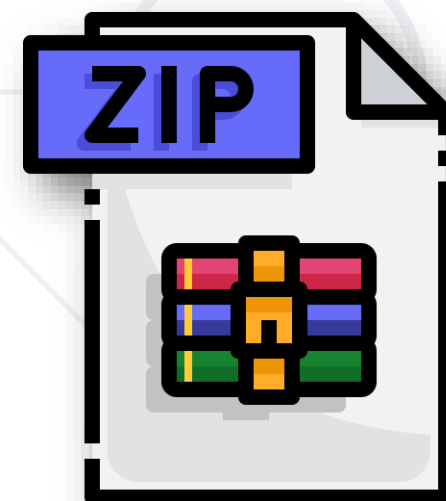
- Тества **функциите**, които една **система трябва да изпълнява**
 - Функциите са **"какво"** трябва да прави **системата**
- **Потвърждава дали** софтуерната система отговаря на **функционалните изисквания**
- Функционалното тестване основно включва тестване тип **черна кутия (black box testing)**
- Функционално **покритие**:
 - Начин да се измери покритието на функционалните тестове върху определена функционалност

Цели на функционалното тестване

- Тестване на **основните функции** на приложение
- **Съобщения за грешка**
 - Проверка дали се извеждат подходящи съобщения за грешка
- **Базисно приложение**
 - Безпроблемна навигация през различните екрани
- **Достъп**
 - Проверка на достъпа на потребителя до системата



- Функционални тестове за файлов архиватор (като 7-Zip или WinRAR):
 - Архивиране на папка в архив
 - Разархивиране на папка
 - Архивиране на един файл
 - Архивиране на няколко файла
 - Архивиране на празна папка
 - и т.н.





Нефункционално тестване

Тества аспекти, които не са функционалност

- **Сешоар: нефункционално тестване**
 - Тест за прегряване: ако сешоара работи 30 минути
 - Колко време отнема да се промени температурата на въздушния поток?
 - Тест за шум
 - Тест за падане
 - Тест за тегло / размери
 - Електрическият кабел достатъчно дълъг ли е?
 - Лесно ли се използва с лява и дясна ръка?

- Ненфункционалното тестване **оценява**:
 - **Надеждност**
 - **Ефективност на работата**
 - **Сигурност / безопасност**
- Тества **"Как"** или **какво е качеството**, с което системата изпълнява своите функции

Цели на нефункционалното тестване

- Нефункционалното тестване се фокусира главно върху **подобряване качеството** на:
 - **Лесната употреба**
 - **Ефективността**
 - **Поддръжката**
 - **Преносимостта на продукта**

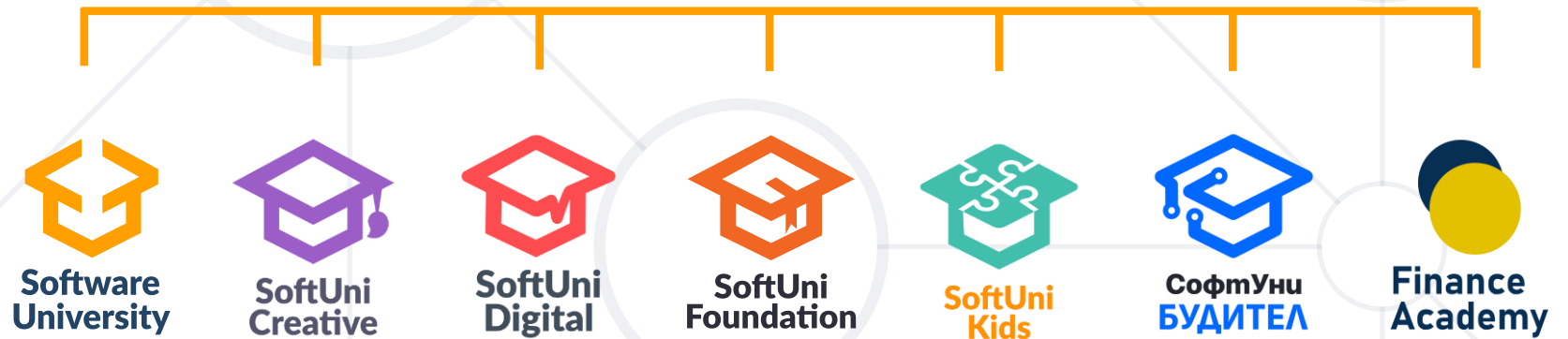


- **Нефункционални тестове за файлов архиватор (7-Zip или WinRAR):**
 - Тест за скорост: колко бързо се компресират файлове / папки
 - Тест за скорост: колко бързо се разархивират файлове / папки
 - Размер на архива: сравнение на различни нива на компресия
 - Тест за сигурност: архивиране / разархивиране на файл, защитен с парола
 - Тест за препълване: компресиране / разархивиране на папка с 500 хил. файла
 - Тест за претоварване: компресиране / разархивиране на 50 файла паралелно

- Различни **нива** на тестване
 - **Юнит тестване**: тестване на единичен компонент
 - **Интеграционно тестване**: тестване на взаимодействието между компонентите
 - **Системно тестване**: QA-те тестват цялата система
 - **Приемно тестване**: Клиентът тества крайния продукт
- Различни **типове** тестване
 - **Функционално тестване**: тестване на софтуерната функционалност
 - **Нефункционално тестване**: производителност, надеждност и др.



Въпроси?



Диамантени партньори



- Software University – High-Quality Education, Profession and Job for Software Developers

- softuni.bg, about.softuni.bg

- Software University Foundation

- softuni.foundation

- Software University @ Facebook

- facebook.com/SoftwareUniversity



- Този курс (презентации, примери, демонстрационен код, упражнения, домашни, видео и други активи) представлява **защитено авторско съдържание**
- Нерегламентирано копиране, разпространение или използване е незаконно
- © СофтУни – <https://softuni.org>
- © Софтуерен университет – <https://softuni.bg>

