



ВЪВЕДЕНИЕ В ТЕСТВАНЕТО

проф. д-р Десислава Петрова-Антонова

Съдържание

- ❖ Тестване: концепция и процес
- ❖ Въпроси, свързани с тестването
- ❖ Основни техники за тестване

Тестване: Защо?

❖ Предназначение

- Осигуряване на софтуерна система, която **работи спрямо очакванията** при използването ѝ от крайните потребители

❖ Тестване на софтуер като краен продукт от производствен процес

- Контролирано експериментиране посредством изпълнение на програмен код

❖ Цел на тестването

- Доказване на качество или правилно поведение
- Установяване и отстраняване на проблеми

❖ Обобщение

- Основно предназначение на тестването е да осигури продукт, в който дефектите отстранени дотолкова, доколкото позволява бюджета или средата

Основни тестови дейности. Тестов процес

❖ Тестови дейности

- Тестово планиране и подготовка
- Изпълнение на тестове и наблюдение
- Анализ и проследяване

❖ **Базов тестов процес** – инстанция на процеса по осигуряване на качество



Особености на тестовите дейности

❖ Минимални тестови дейности

- Изпълнение на софтуера и наблюдение
- Неформално vs. формално тестване

❖ Ефективност на тестовото изпълнение

- Гарантиране, че неуспешните тестови сценарии няма да блокират изпълнението на останалите
- Премахване на дефекти
 - ✓ Може да се отнесе и към процеса на разработка на софтуер
- Повторна верификация

❖ Анализ на данните, получени от тестовото изпълнение

- Локализиране и премахване на дефекти
- Получаване на обратна връзка за тестовия процес и цялостния процес на разработка
- Идентифициране на възможности за дългосрочно подобряване на качеството на продукта

Поддейности на тестовото планиране и подготовка

- ❖ Необходимост от планиране и подготовка
 - Систематизирано покритие на продуктовите характеристики и функционалност
- ❖ Определяне на тестова цел
 - Постигане на надежност или определено тестово покритие като критерий за изход от тестовия процес
- ❖ Дефиниране на тестови сценарии
 - Конструирание, автоматично генериране или избор на тестови сценарии въз основа на **формални модели**, свързани с **формалните техники за тестване**
- ❖ Дефиниране на тестова процедура
 - **Формализирането** на тестовата процедура е условие за постигане на ефективност при изпълнението и управлението на тестовия процес

Тестването като част от управление на качеството

❖ Тестване & Инспекция

- Тестването е **средство за редукция на дефекти** подобно на инспекцията и статичния и динамичния анализ
- Тестването открива дефектите **индиректно** посредством изпълнението на програмен код
- Тестването и инспекцията откриват **различни типове** проблеми

❖ Тестване & Превенция от грешки

- Превенцията от грешки **редуцира дефектите**, които се отстраняват на етап тестване

❖ Тестване & Формална верификация

- Формалната верификация **спестява усилия**, необходими при тестването на базовата функционалност на продукта

❖ Тестване & Стратегии за устойчивост на грешки и ограничаване на повреди

- Стратегиите за устойчивост на грешки и ограничаване на повреди се прилагат при **критични системи**, работещи в среди с трудни за предсказване събития

Тестването като част от процеса на разработване

❖ Разработване на софтуер при следване на модела на “водопада”

- В по-голямата си част тестването е обособено като отделна фаза от разработването на софтуера
- Разработването на софтуера е съпроводено от тестване на всяко ниво
- Анализът на тестовите резултати продължава и след приключване на тестовите дейности

❖ Разработване на софтуер при използване на методологията за екстремно програмиране

- Тестването е част от процеса на разработване на софтуера (test-driven development)

Базови въпроси, свързани с тестването

- ❖ Какви артефакти се тестват?
 - **Софтуерни програми и код**, написани на различни програмни езици
- ❖ Какво се тества и какъв тип дефекти се откриват?
 - Функционално тестване
 - ✓ Поведението на софтуера отговаря ли на потребителските очаквания или продуктовата спецификация?
 - Структурно тестване
 - ✓ Правилно ли са реализирани вътрешните програмни единици, структури и връзките между тях?
- ❖ Кога и на какво ниво на дефектите тестването трябва да спре?
 - **Ниво на покритие** като критерий за край на тестовия процес
 - ✓ Гарантира премахването на определен тип дефекти на ниско ниво, при което качеството на продукта не се оценява директно
 - **Ниво на надеждност** като критерий за край на тестовия процес
 - ✓ Гарантира отстраняване на дефекти, които с най-голяма вероятност предизвикват проблеми при потребителите (тестване, базирано на употреба)

Въпроси, свързани с техниките за тестване 1/3

- ❖ Каква **техника за тестване** да се използва за конкретна софтуерна система?
 - Изборът на техника се влияе от предмета на тестване и от критериите за край на тестовия процес
- ❖ Какъв е **моделът, използван при определена техника за тестване?**
 - Модели, използващи прости структури
 - ✓ Контролни списъци
 - ✓ Класове на еквивалентност
 - Модели, използващи сложни структури
 - ✓ Машина на крайните състояния

❖ Какъв е **моделът**, използван при определена техника за тестване?

- Използване на модели за определяне на тестовото покритие
- Използване на модели при тестването, базирано на употреба
 - ✓ Добавяне на вероятности за използване към елементите на формалните модели
 - Оперативен профил
 - Верики на Марков
- Разширение на моделите с цел поддръжка на широко използвани техники за тестване
 - ✓ Тестване на входния домейн (разширение на тестването с класове на еквивалентност)
 - ✓ Тестване на управляващия поток и потока от данни (разширение на тестването с машина на крайните състояния)

Въпроси, свързани с техниките за тестване 3/3

- ❖ Приложими ли са техниките за тестване от **други домейни** при тестването на софтуерни системи?
 - Посяването на грешки, мутациите и имунизациите са техники използвани във физическите, биологичните и социалните системи
- ❖ Могат ли да се **комбинират или интегрират различни техники** за тестване с цел постигане на по-висока ефективност и производителност?
 - Различните техники могат да споделят общи идеи, модели и артефакти, поради което могат да се използват съвместно с цел повишаване на качеството на продукта и минимизиране на цената и усилията при тестване

Въпроси, свързани с дейностите и управлението

- ❖ **Кой изпълнява** специфични тестови дейности?
- ❖ **Кога могат да бъдат изпълнени** специфични тестови дейности?
 - Наличние на програмните единици, компоненти и системи, подлежащи на тестване
 - Стартиране на специфична фаза от разработването на софтуера
 - Специфично тестване на специфичен продукт или специфична ситуация
- ❖ **Какъв е процесът**, който се следва при изпълнението на тестовите дейности?
- ❖ Възможно ли е **автоматизиране** на тестовите дейности?
 - Какви инструменти за тестване са налични и приложими за конкретен продукт?
- ❖ **Какви артефакти се използват за управление** на тестовия процес?
- ❖ **Каква е хардуерната, софтуерната и организационната среда** за тестване?
- ❖ **Какъв е продуктовият тип и пазарен сегмент** на тестваната софтуерна система?
 - Специфични техники за тестване vs. Техники за тестване с общо предназначение

Какво да тестваме?



Обект на тестването

❖ Нива на абстракция

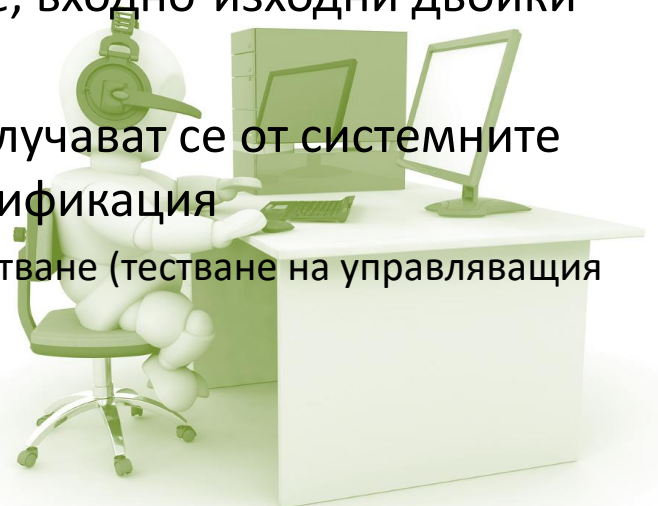
- Тестване на **отделни програмни елементи** (функция, процедура, метод в зависимост от използвания език за разработка)
- Тестване на **група от свързани програмни елементи или компоненти**
- Тестване на **системата като цяло** от гледна точка на нейната функционалност и входно-изходни операции

❖ Колкото **по-високо е нивото на абстракция**, толкова по-приложимо е **функционалното тестване** и обратно – колкото **по-ниско е нивото на абстракция**, толкова по-приложимо е **структурното тестване**

❖ Нивата на абстракция определят фазите на тестовия процес (тестване на ниво програмна единица, компонентно тестване, тестване на интеграция, системно тестване, тестване за приемане на системата, бета тестване и т.н.)

Функционално тестване

- ❖ Разглежда софтуера като **черна кутия**
- ❖ Реализира се посредством **наблюдение на външното поведение на софтуера** при изпълнението му
- ❖ Форми на функционално тестване
 - “**ad hoc**” тестване: разчита се, че неочакваното поведение на системата лесно може да се разграничи от очакваното
 - Използване на **контролен списък**: списък с външни функции, информация за очакваното поведение, входно-изходни двойки (пример калкулатор)
 - Използване на **формални модели**: получават се от системните изисквания или функционалната спецификация
 - ✓ Адаптиране на техники от структурното тестване (тестване на управляващия поток и потока от данни)



Изпълнение на функционалното тестване

❖ Следване на базовия тестов процес

■ Планиране

- ✓ Определяне на външните функции, входните условия и очаквания изход
- ✓ Пример компилатор
 - Тестване с коректни и некоректни програми
 - Тестване на време за приключване на компилацията

■ Изпълнение

- ✓ Наблюдение на външното поведение на системата
- ✓ Запис на получената при изпълнението информация за последващ анализ
 - Получен резултат
 - Последователност на изпълнение
 - Среда на изпълнение

■ Анализ

- ✓ Сравняване на действителното поведение с очакваното
- ✓ Изпълнение на релевантни дейности
 - Пресъздаване на ситуации на повреда
 - Диагностициране на проблеми
 - Отстраняване на дефекти
 - Вземане на решение за развитието на тестовия процес

Структурно тестване

- ❖ Разглежда софтуера като **бяла кутия**
- ❖ Верифицира правилната реализация на вътрешните системни единици и взаимовръзките между тях
- ❖ Форми за структурно тестване
 - Дебъгване: проследяване на изпълнението на програмните изрази
 - ✓ Преимущество: Проблемите се откриват и заедно с това се локализируют
 - ✓ Недостатък: Трудно се откриват пропуски, допуснати при проектирането
 - ✓ Обикновено се извършва от разработчиците на софтуер
 - Използване на **формални модели**: получават се от информацията за програмната реализация на системата
 - Следване на **базовия тестов процес**
 - ✓ Изпълнява се в малък мащаб поради необходимостта от задълбочени познания за програмната реализация и опасността от комбинаторно нарастване на системните единици, които трябва да бъдат покрити
 - ✓ По-малка необходимост от формализация, по-малка роля на планирането, по-лесно локализиране на дефекти, по-лесно дефиниране на критерии за спиране

Структурно тестване vs. Функционално тестване

Критерии	Структурно тестване	Функционално тестване
Перспектива	Системата се разглежда като бяла кутия	Системата се разглежда като черна кутия
Обект на тестването	Тестват се малки програмни единици и системни компоненти	Тества се системата като цяло
График на изпълнение	Прилага се на по-ранните фази от разработката на системата	Прилага се на по-късните фази от разработката на системата
Типове дефекти	Дефекти, свързани с програмната реализация на системата	Дефекти, свързани с външните функции на системата
Откриване на дефекти	Откриват се по-лесно, отстраняват се проблеми в малки програмни единици	Откриват се по-трудно, отстраняват се интерфейсни и комуникационни проблеми
Техники за тестване	Прилагат се както специфични, така и общи техники за тестване	
Тестер	Обикновено се изпълнява от разработчиците	Обикновено се изпълнява от професионални тестери или външни организации

Кога трябва да спре тестовият процес?



Критерии за спиране на тестването 1/2

❖ Неформални критерии

- Критерии, базирани на ресурси
 - ✓ Липса на време
 - ✓ Липса на средства
- Критерии, базирани на дейности
 - ✓ Завършване на планираните тестови дейности

❖ Критерий за постигане на надеждност

- Надеждността се оценява въз основа на тестовите данни и се сравнява с дефинираната при планиране на качеството
- Оценката на надеждността се сравнява с очакваната от потребителите
 - ✓ Тестване базирано на употреба, статистическо тестване
 - ✓ Неприложимост при ранните фази на тестване

Критерии за спиране на тестването 2/2

- ❖ Постигане на надеждност при ранните фази на тестване
 - Наличие на компоненти, които не се използват директно от крайния потребител
 - Наличие на критични компоненти с ниска честота на използване
- ❖ „Продуктът не може да се пуска в експлоатация, докато всички компоненти не са тествани“
- ❖ Критерий за постигане на тестово покритие
 - Тестовото покритие е различно за различните техники на тестване
 - Допуска се, че покритите системни единици са без дефекти
 - Допуска се, че по-високото тестово покритие води до по-високо качество на продукта
 - **Препоръчва се да се използва, когато липсва оценка за надеждността**

Статистическо тестване, базирано на употреба (СТУ)

- ❖ Проблем: откриването на дефекти при употребата на продукта от крайните потребители е полезна, но скъпа процедура
- ❖ Решение: изпълнение на т. нар. бета тестове или статистическо тестване, базирано на употреба
 - Тестовата среда имитира продуктовата среда
 - Изпълнението на тестовите сценарии следва начина и шаблоните на използване на системата от крайните потребители
 - Многобройните потребители и шаблони на използване се редуцират статистически
- ❖ Реализация: използване на оперативен профил
 - Плоски оперативни профили
 - Оперативни профили на Марков
- ❖ Приложение: финална фаза на тестването
 - Тестване за приемане на системата преди пускането на продукта за употреба
- ❖ Критерий за спиране: постигане на определена степен на надеждност

Тестване, базирано на покритие (ТП)

- ❖ Същност: покриване на всички елементи във формалния модел, използван от съответната техника за тестване
 - Контролен списък
 - Формално дефинирани класове
 - Разделяне на домейни в поддомейни и определяне на гранични условия
 - Машини на крайните състояния
- ❖ Последователност на изпълнение
 - Дефиниране на модел
 - ✓ Представяне с граф
 - Проверка на елементите от модела
 - ✓ Установяване, че елементите са тествани самостоятелно
 - ✓ Цялостна валидация на модела
 - Дефиниране на критерий за покритие
 - ✓ Покриване на елементите от модела и използване на допълнителни критерии за покритие (покритие на границите)
 - Създаване на тестови сценарии
 - ✓ Определяне на вход, изход и начин за проверка на резултата

СТУ vs. ТП

Критерии	СТУ	ТП
Перспектива	Обектите се разглеждат от гледна точка на потребителя	Обектите се разглеждат от гледна точка на разработчика
Критерии за спиране	Постигане на надеждност	Постигане на покритие
Обект	Обикновено се прилага върху системата като цяло	Обикновено се прилага при малки обекти
Верификация vs. Валидация	По-подходящо е за извършване на валидиращи тестове	По-подходящо е за извършване на верифициращи тестове
График на изпълнение	Прилага се на по-късните фази от тестването	Прилага се на по-ранните фази тестването
Откриване на дефекти	Откриват се дефекти с потенциал за поява при потребителите с което се повишава надеждността на системата	Ефективно се откриват дефекти, които не винаги са свързани с надеждността на системата
Тестова среда	Средата е близка до инсталираната при крайния потребител	Средата е специално инсталирана за тестови цели
Техники	Могат да използват общи техники и формални модели	
Роля на потребителите	Моделите се конструират със съдействието на потребителите	Моделите не изискват активно съдействие от потребителите
Тестер	Изпълнява се от професионални тестери	Изпълнява се от професионални тестери и програмисти

