



CS619

КОМПОНЕНТНО ТЕСТВАНЕ НА СОФТУЕР

Приета: прот. № 15 / 01.10.2020 г.

Актуализирана: прот. № 29 / 29.06.2023 г.

Лектор: доц. д-р Димитър Минчев

АНОТАЦИЯ

Дисциплината дава основни знания и умения за верификация на софтуер по време на разработка му посредством използването на компоненти тестове. Разглеждат се понятията: писане на компоненти тестове, анализиране на покритието на кода с тестове, избягване на зависимости, имитиране на обекти.

ОСНОВНИ ЦЕЛИ

Основна цел на дисциплината е да се придобият знания и умения за верификация на софтуер по време на разработка му посредством използването на компоненти тестове. След успешното приключване на дисциплината, студентите ще могат да:

- Пишат компонентни тестове,
- Анализират покритието на кода с тестове,
- Избягват зависимости,
- Имитират обекти.

ПРЕДПОСТАВКИ

Необходима предпоставка за дисциплината е предварителна подготовка на студентите по програмиране, обектноориентирано програмиране и бази данни.

СТАТУТ И СТРУКТУРА

специалност	статут	Кредити	редовно обучение				задочно обучение			
			л	с	у	общ	л	с	у	общ
СИ	Задължителна	6	30	30	-	60	15	15	-	30
ПП	Задължителна	6	30	30	-	60	15	15	-	30

СЪДЪРЖАНИЕ НА КУРСА

1. **Unit tests.** NUnit vs. Microsoft Visual Studio Team Tests (VSTT).
2. **AAA.** Arrange. Act. Assert.
3. **Attributes.** TestClass. TestMethod. ExpectedException. Timeout. Ignore. ClassInitialize, ClassCleanup. TestInitialize, TestCleanup.
4. **Asserts.** AreEqual. AreSame. IsNull. IsNotNull. IsTrue. IsFalse. Fail.
5. **Dependencies.** Avoid using Interfaces.
6. **MOQ.** KickStart Unit Testing.

ПЛАНИРАНИ УЧЕБНИ ДЕЙНОСТИ И МЕТОДИ НА ОБУЧЕНИЕ

Методи на обучение

- Присъствени лекции и семинарни упражнения
- Практическо обучение
- Интерактивно обучение
- Визуално обучение
- Електронно обучение чрез платформите Moodle и Microsoft Teams

Средства на обучение

- Самостоятелна работа
- Практически задачи
- Решаване на задачи с използване на приложен софтуер
- Учебни видео материали включително видео презентации
- Използване на електронни ресурси в платформите Moodle и Microsoft Teams под формата на теоретични материали, тестове и задачи за самостоятелна работа към всяка тема

МЕТОДИ ЗА ОЦЕНЯВАНЕ

Работата на всеки студент през семестъра се оценява в точки. Те се получават в резултат от аудиторна и извън аудиторна заетост на студента, както и от изпитна процедура, която представлява тест с въпроси от отворен и затворен тип. Максималният брой точки които студента може да събере са 100, а разпределението им по оценяваните дейности е следното:

- | | |
|----------------------------------|----------|
| 1. Аудиторна заетост..... | 20 точки |
| 2. Извън аудиторна заетост | 30 точки |
| 3. Изпитна процедура..... | 50 точки |

Окончателната оценка се формира като точките се разпределят по скалата:

- от 36 до 50 точки = Среден (3);
- от 51 до 65 точки = Добър (4);
- от 66 до 80 точки = Много добър (5);
- от 81 до 100 точки = Отличен (6).

ПРЕПОРЪЧИТЕЛНА ЛИТЕРАТУРА

1. James W. Newkirk, Alexei A. Vorontsov. Test-Driven Development in Microsoft .NET (Developer Reference). Microsoft Press. 2004.
2. Roy Oshero. The Art of Unit Testing: With Examples in .net. Manning Publications. 2009.
3. Vladimir Khorikov. Unit Testing Principles, Practices, and Patterns. Manning Publications. 2020.

ИНТЕРНЕТ РЕСУРСИ

4. Unit testing in .NET Core and .NET Standard: <https://docs.microsoft.com/bg-bg/dotnet/core/testing/?pivots=mstest>
5. Microsoft Unit Testing: <https://docs.microsoft.com/en-us/visualstudio/test/getting-started-with-unit-testing?view=vs-2019>
6. Обучителен модул 5 "Обектно-ориентирано програмиране" от национална програма "Обучение за ИТ кариера" на МОН за професия "Приложен програмист":
<https://github.com/dimitarminchev/ITCareer/tree/master/05.%20Object%20Oriented%20Programming>
и <https://github.com/dimitarminchev/ITCareer/wiki/05.-Обектно-ориентирано-програмиране>
7. Обучителен модул 7 "Разработка на софтуер" от национална програма "Обучение за ИТ кариера" на МОН за професия "Приложен програмист":
<https://github.com/dimitarminchev/ITCareer/tree/master/07.%20Software%20Development> и
<https://github.com/dimitarminchev/ITCareer/wiki/07.-Разработка-на-софтуер>
8. MOQ Quickstart, <https://github.com/Moq/moq4/wiki/Quickstart>
9. C# Writing unit tests with NUnit and Moq, <https://developerhandbook.com/unit-testing/writing-unit-tests-with-nunit-and-moq/>
10. KickStart your Unit Testing using Moq: <https://www.codeproject.com/Articles/796014/KickStart-your-Unit-Testing-using-Moq>