

АННОТАЦИЯ К РАБОЧЕЙ ПРОГРАММЕ ДИСЦИПЛИНЫ  
«Тестирование программного обеспечения автоматизированных систем»

по основной профессиональной образовательной программе по направлению подготовки  
09.03.01 «Информатика и вычислительная техника» (уровень бакалавриата)

**Направленность (профиль):** Программно-техническое обеспечение автоматизированных систем  
**Общий объем дисциплины – 4 з.е. (144 часов)**

**Форма промежуточной аттестации – Зачет.**

**В результате освоения дисциплины у обучающихся должны быть сформированы компетенции с соответствующими индикаторами их достижения:**

- ПК-7.1: Демонстрирует знание технологий тестирования;
- ПК-7.2: Осуществляет тестирование и анализирует результаты тестирования программного обеспечения;
- ПК-7.3: Использует специальное программное обеспечение для автоматизированного тестирования;

**Содержание дисциплины:**

Дисциплина «Тестирование программного обеспечения автоматизированных систем» включает в себя следующие разделы:

**Форма обучения очная. Семестр 6.**

**1. Основы информатики и программирования в сфере тестирования программ.**

**Необходимость процессов тестирования и верификации программного обеспечения.**

**Специфицирование программного обеспечения и его тестирование методом черного ящика.**

**Роли людей, процессов, методов, инструментов и технологий обеспечения качества..**

Формализация задач. Алгоритмическое решение. Тестирование и верификация, основные определения. Тестировщик или QA-инженер.

Тестирование черного и белого ящика. Регрессивное тестирование. Процесс тестирования и V-модели. Спецификация в виде инвариантов, предусловий и постусловий. Спецификация в виде Use-Case диаграмм языка UML. Работа тестировщика. Работа с баг-трекинговой системой. Краткая справка по командной работе на GitHub..

**2. Концепции и атрибуты качества на уровне кода. Отладка кода. Модульное тестирование и документирование проекта.**

Отладка кода и отладчики. Место процесса модульного тестирования в процессе разработки. Документирование проекта на уровне кода. Обеспечение надёжности в процессе модульного тестирования. Фреймворки модульного тестирования. Некоторые советы по модульному тестированию. Анализ результатов тестирования..

**3. Применение основных методов разработки в тестировании. Методологии разработки \*DD (TDD, BDD, MDD). Разработка через тестирование. Конструирование программ на основе предварительных тестов.** Применение основных методов и инструментов разработки прямо в процессе тестирования.

Что такое методология разработки. MDD (Model-Driven Development). Разработка через тестирование (TDD, Test-Driven Development). Разработка, управляемая поведением (BDD, Behavior-Driven Development). Настройка инструментов и примеры..

**4. Функциональное автоматизированное тестирование.** Зачем нужно функциональное тестирование и зачем его автоматизировать. Подходы к автоматизации. Тестирование настольных (desktop) приложений с помощью IBM Rational Functional Tester. Тестирование Web-приложений с помощью Selenium. Разработка по BDD с помощью Selenium WebDriver..

**5. Методы анализа программ и их верификации. Статические проверки и динамический анализ программ. Логика Флойда-Хоара. Тройки Хоара. Дедуктивная верификация.**

**Контракты на код..** Статический анализ, его необходимость. SonarQube. PVS-Studio. Cppcheck. О написании собственных статических анализаторов. Динамический анализ с помощью Valgrind. Дедуктивная верификация. Контракты на код. Язык программирования Eiffel. Design By Contract. Microsoft .NET Code Contracts..

Разработал:

доцент  
кафедры ПМ

С.М. Старолетов

Проверил:  
Декан ФИТ

А.С. Авдеев