

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ
ФЕДЕРАЦИИ

Федеральное государственное бюджетное образовательное учреждение
высшего образования

«КУБАНСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ»

Физико-технический факультет



УТВЕРЖДАЮ:

Проректор по учебной работе,
качеству образования – первый
проректор

Т.А. Хагуров

«30» мая 2025 г.

РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ (МОДУЛЯ)

Б1.В.ДВ.03.02 Методы и средства тестирования программного обеспечения

)

Направление подготовки/специальность

09.04.02 Информационные системы и технологии

Направленность (профиль) / специализация

Системы и сети доставки цифрового контента

Форма обучения

очно-заочная

Квалификация

магистр

Краснодар 2025

Рабочая программа дисциплины составлена в соответствии с федеральным государственным образовательным стандартом высшего образования (ФГОС ВО) по направлению подготовки / специальности

Программу составил(и):

Рабочая программа дисциплины на заседании кафедры ИФ и КТ
протокол № 9 от «08» 04 2025 г.

Заведующий кафедрой д. физ.-мат. наук,
профессор К.А. Лебедев.



Утверждена на заседании учебно-методической комиссии
факультета/института УМК ФТФ №11 от 21.04. 2025 г
Председатель УМК факультета/института

д-р физ. мат. наук, профессор

Богатов Н. М.


ПОДПИСЬ

Рецензенты:

Абрамов Д. Е. канд. хим. наук директор ООО «Ресурс»

Шевченко А. В. канд. физ-мат. наук. Ведущий специалист ООО «Южная аналитическая компания»

2. Аннотация

Дисциплина "Методы и средства тестирования программного обеспечения" формирует у магистрантов системное представление о современных подходах к обеспечению качества программного обеспечения через комплексное тестирование. Курс охватывает полный спектр методов тестирования - от модульного и интеграционного до системного и приемочного, с акцентом на автоматизацию процессов тестирования.

Особое внимание уделяется методологиям тест-дизайна, созданию автоматизированных тестовых фреймворков, непрерывному тестированию в CI/CD, а также метрикам и управлению качеством ПО на основе данных тестирования.

3. Место дисциплины в структуре ОПОП

Пререквизиты:

- "Технологии разработки программного обеспечения"
- "Объектно-ориентированное программирование"
- "Базы данных"
- "Операционные системы"

Последующие дисциплины:

- "Управление IT-проектами"
 - "DevOps-практики и CI/CD"
 - "Проектирование и архитектура программных систем"
 - "Магистерская диссертация"
-

4. Планируемые результаты обучения

Дисциплинарные компетенции (ДК):

В результате освоения дисциплины студент должен:

Знать:

- Фундаментальные принципы и виды тестирования ПО
- Методологии тест-дизайна и техники тестирования
- Процессы и жизненный цикл тестирования
- Методы автоматизации тестирования
- Метрики и оценку качества ПО

- Современные инструменты и фреймворки тестирования

Уметь:

- Разрабатывать тестовую стратегию и планы тестирования
- Создавать тестовые сценарии и тест-кейсы
- Автоматизировать процессы тестирования
- Анализировать результаты тестирования и отчеты
- Интегрировать тестирование в CI/CD процессы
- Оценивать покрытие кода и эффективность тестирования

Владеть:

- Навыками работы с инструментами автоматизации тестирования
 - Методами тест-дизайна и создания тестовой документации
 - Технологиями модульного, интеграционного и системного тестирования
 - Практиками тестирования производительности и безопасности
 - Инструментами анализа покрытия кода и метрик качества
-

5. Содержание дисциплины

Модуль 1: Основы тестирования ПО

- 1. Фундаментальные принципы тестирования**
 - Цели и задачи тестирования ПО
 - Принципы тестирования (7 фундаментальных принципов)
 - Основные понятия: ошибки, дефекты, сбои
 - Психология тестирования
- 2. Процесс тестирования**
 - Жизненный цикл тестирования
 - Тест-анализ и проектирование тестов
 - Реализация и выполнение тестов
 - Оценка критериев выхода и отчетность
- 3. Уровни тестирования**
 - Модульное тестирование (Unit Testing)
 - Интеграционное тестирование
 - Системное тестирование
 - Приемочное тестирование

Модуль 2: Методы тест-дизайна

- 1. Техники тестирования черного ящика**
 - Эквивалентное разделение

- Анализ граничных значений
- Таблицы решений
- Тестирование переходов между состояниями
- 2. Техники тестирования белого ящика**
 - Покрытие операторов
 - Покрытие решений
 - Покрытие условий
 - Покрытие путей
- 3. Специализированные виды тестирования**
 - Тестирование производительности (Load, Stress, Volume)
 - Тестирование безопасности
 - Тестирование удобства использования (Usability)
 - Тестирование совместимости

Модуль 3: Автоматизация тестирования

- 1. Фреймворки автоматизации**
 - Принципы автоматизации тестирования
 - Архитектура тестовых фреймворков
 - Page Object Pattern и другие паттерны
 - Критерии выбора инструментов автоматизации
- 2. Инструменты тестирования**
 - Фреймворки модульного тестирования (JUnit, TestNG, pytest)
 - Инструменты веб-тестирования (Selenium, Cypress)
 - API тестирование (Postman, RestAssured)
 - Мобильное тестирование (Appium)
- 3. Непрерывное тестирование**
 - Интеграция тестирования в CI/CD
 - Тестирование в DevOps-процессах
 - Мониторинг качества в реальном времени

Модуль 4: Управление качеством и метрики

- 1. Управление тестированием**
 - Оценка и планирование тестирования
 - Мониторинг и контроль тестирования
 - Управление тестовой документацией
 - Управление дефектами
- 2. Метрики качества**
 - Метрики покрытия кода
 - Метрики эффективности тестирования
 - Метрики качества дефектов
 - Оценка зрелости процессов тестирования
- 3. Современные тенденции**
 - TestOps и AI в тестировании

- Тестирование в Agile и DevOps
 - Shift-left подходы
 - Тестирование микросервисных архитектур
-

6. Образовательные технологии

- **Практико-ориентированные занятия** с реальными проектами
 - **Лабораторные работы** с современными инструментами тестирования
 - **Кейс-стади** из промышленной практики
 - **Мастер-классы** от ведущих QA-инженеров
 - **Проектная работа** в командах
 - **Парное программирование** и ревью тестов
-

7. Оценочные средства

Текущий контроль (40%):

- **Лабораторные работы (25%):**
 - Разработка модульных тестов
 - Создание автоматизированных UI-тестов
 - Тестирование REST API
 - Нагрузочное тестирование
- **Практические задания (15%):**
 - Разработка тест-плана и тест-кейсов
 - Анализ покрытия кода
 - Ревью тестовой документации

Рубежный контроль (60%):

- **Курсовой проект (35%):**
 - Разработка комплексной системы тестирования для реального приложения
 - Создание автоматизированного тестового фреймворка
 - Интеграция тестирования в CI/CD конвейер
 - Защита проекта с демонстрацией результатов
 - **Экзамен (25%):**
 - Теоретическая часть (методологии, принципы, процессы)
 - Практический кейс (разработка тестовой стратегии)
-

8. Учебно-методическое обеспечение

Основная литература:

1. Савин Р. "Тестирование Dot Com"
2. Куликов С.В. "Автоматизация функционального тестирования"
3. Котляров В.П. "Основы тестирования программного обеспечения"

Дополнительная литература:

1. Myers G.J., Sandler C., Badgett T. "The Art of Software Testing"
2. Selenium WebDriver documentation
3. JUnit 5 User Guide

Программное обеспечение:

- Фреймворки тестирования: JUnit, TestNG, pytest
- Инструменты веб-тестирования: Selenium WebDriver, Cypress
- API тестирование: Postman, RestAssured
- Системы CI/CD: Jenkins, GitLab CI
- Инструменты анализа: JaCoCo, SonarQube

Онлайн-ресурсы:

- Официальная документация инструментов тестирования
- ISTQB учебные материалы
- Онлайн курсы по автоматизации тестирования
- Профессиональные сообщества QA-инженеров

9. Материально-техническое обеспечение

- Компьютерные классы с установленным ПО для тестирования
- Лицензии на коммерческие инструменты тестирования
- Доступ к облачным средам для тестирования
- Лабораторные стенды для нагрузочного тестирования
- Мобильные устройства для кроссплатформенного тестирования