

Федеральное государственное бюджетное образовательное учреждение высшего образования
«Алтайский государственный технический университет им. И.И. Ползунова»

СОГЛАСОВАНО

Декан ФИТ

А.С. Авдеев

Рабочая программа дисциплины

Код и наименование дисциплины: **Б1.В.10 «Тестирование программного обеспечения автоматизированных систем»**

Код и наименование направления подготовки (специальности): **09.03.01 Информатика и вычислительная техника**

Направленность (профиль, специализация): **Программно-техническое обеспечение автоматизированных систем**

Статус дисциплины: **часть, формируемая участниками образовательных отношений**

Форма обучения: **заочная**

Статус	Должность	И.О. Фамилия
Разработал	старший преподаватель	Я.Б. Ерёмин
Согласовал	Зав. кафедрой «ПМ»	Е.Г. Боровцов
	руководитель направленности (профиля) программы	Л.И. Сучкова

г. Барнаул

1. Перечень планируемых результатов обучения по дисциплине, соотнесенных с индикаторами достижения компетенций

Компетенция	Содержание компетенции	Индикатор	Содержание индикатора
ПК-7	Способен осуществлять тестирование программного обеспечения, в том числе с применением автоматизированных средств	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

2. Место дисциплины в структуре образовательной программы

Дисциплины (практики), предшествующие изучению дисциплины, результаты освоения которых необходимы для освоения данной дисциплины.	Программирование
Дисциплины (практики), для которых результаты освоения данной дисциплины будут необходимы, как входные знания, умения и владения для их изучения.	Преддипломная практика

3. Объем дисциплины в зачетных единицах с указанием количества академических часов, выделенных на контактную работу обучающегося с преподавателем (по видам учебных занятий) и на самостоятельную работу обучающегося

Общий объем дисциплины в з.е. /час: 4 / 144

Форма промежуточной аттестации: Зачет

Форма обучения	Виды занятий, их трудоемкость (час.)				Объем контактной работы обучающегося с преподавателем (час)
	Лекции	Лабораторные работы	Практические занятия	Самостоятельная работа	
заочная	6	10	0	128	21

4. Содержание дисциплины, структурированное по темам (разделам) с указанием отведенного на них количества академических часов и видов учебных занятий

Форма обучения: заочная

Семестр: 7

Лекционные занятия (6ч.)

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. {беседа} (2ч.)[1,2,13,14] 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 {лекция-пресс-конференция} (2ч.)[1,2,5,7,8,11,15,16,17] 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. Функциональное автоматизированное тестирование {лекция-пресс-конференция} (2ч.)[1,2,4,6,9,10,11,12] Зачем нужно функциональное тестирование и зачем его автоматизировать. Подходы к автоматизации. Тестирование настольных (desktop) приложений с помощью Lanit NeuroControl. Тестирование Web-приложений с помощью Selenium и Postman.

Лабораторные работы (10ч.)

1. Создание спецификации для существующего ПО в виде сценариев использования(2ч.)[1,2] Разработка спецификации на существующее приложение, например курсовой проект студента. Спецификация в виде сценариев использования.

2. Создание спецификации для существующего ПО на языке UML(2ч.)[1,13,14] Разработка спецификации на существующее приложение, например курсовой проект студента. Спецификация в виде UML Use Case диаграмм.

3. Создание документации для кода(2ч.)[1,5] Написание документирующих комментариев и генерация с помощью Doxygen или аналогов, вставка графов, изображений и формул.

4. Работа с баг-трекинговой системой, составление баг-репорта. Написание модульных тестов.(2ч.)[1]

5. Функциональное автоматизированное тестирование web API.(2ч.)[1,2,12]

Работа в среде Postman и взаимодействие с API для авторизации, аутентификации, получения данных, их обновления, добавления, и удаления.

Самостоятельная работа (128ч.)

1. Освоение учебной литературы по тестированию(60ч.)[1,2]

2. Изучение литературы в виде пособия, а также предложенных публикаций с целью изучения существующих методов тестирования программного обеспечения.(16ч.)[1,2,9,10,11,12,16] Изучаются преимущества и недостатки методов, а также трудоемкость их осуществления, и делается вывод о применимости данного класса методов для реальных задач для ИВТ.

3. Анализ и рефакторинг (улучшение) своего кода при выполнении лабораторных работ(20ч.)[1,2,8] Предлагать и осуществлять рефакторинг своего ранее реализованного программного обеспечения, так, чтобы возможно было его тестировать автоматизированными методами.

4. Выполнение контрольной работы.(20ч.)[1,2] Необходимо письменно ответить на вопросы, а также, по возможности, продемонстрировать ответ кодом теста. Примеры вопросов:

-Дайте основные определения в сфере тестирования.

-Опишите работу тестировщика или QA-инженера в компании по разработке ПО.

-Распишите отличия в тестирование черного, серого и белого ящика. Для каких типов программных систем применим каждый из методов?

-Обоснуйте необходимость регрессивного тестирования.

-Продемонстрируйте процесс тестирования на примере и опишите V-модели в тестировании.

-Опишите спецификации в виде инвариантов, предусловий и постусловий для вашего примера программного обеспечения.

5. Подготовка к промежуточной аттестации (зачет)(4ч.)[1,2,4,5,6,7,8,9,10,11,12,13,14,15,16,17]

6. Подготовка к защите контрольной работы.(8ч.)[1,2,4,5,6,7,8,9,10,11,12,13,14,15,16,17]

5. Перечень учебно-методического обеспечения самостоятельной работы обучающихся по дисциплине

Для каждого обучающегося обеспечен индивидуальный неограниченный доступ к электронно-библиотечным системам: Лань, Университетская библиотека он-лайн, электронной библиотеке АлтГТУ и к электронной информационно-образовательной среде:

1. Старолетов С.М. Основы тестирования и верификации программного обеспечения [Электронный ресурс]: Учебное пособие.— Электрон. дан.— Барнаул: АлтГТУ, 2020.— Режим доступа:

http://elib.altstu.ru/eum/download/pm/Staroletov_OsnTestVerifPO_up.pdf,
авторизованный

6. Перечень учебной литературы

6.1. Основная литература

2. Старолетов С. М. Автоматное моделирование многокомпонентных программных систем, их тестирование и верификация: учебно-методическое пособие/ С. М. Старолетов, Е. Н. Крючкова. – Барнаул : Изд-во АлтГТУ, 2015. – 156 с. — Режим доступа: http://new.elib.altstu.ru/eum/download/pm/avtomat_staroletov.pdf

6.2. Дополнительная литература

3. Липаев, В. В. Тестирование компонентов и комплексов программ : учебник / В. В. Липаев. — Москва : СИНТЕГ, 2010. — 393 с. — ISBN 978-5-89638-115-0. — Текст : электронный // Цифровой образовательный ресурс IPR SMART : [сайт]. — URL: <https://www.iprbookshop.ru/27301.html> (дата обращения: 05.05.2023). — Режим доступа: для авторизир. пользователей

7. Перечень ресурсов информационно-телекоммуникационной сети «Интернет», необходимых для освоения дисциплины

4. Selenium - Web Browser Automation. — Режим доступа: <https://www.seleniumhq.org>

5. Doxygen. Generate documentation from source code. — Режим доступа: <http://www.doxygen.nl>

6. TDD в геймдеве или «кроличий ад». — Режим доступа: <https://habr.com/ru/companies/pixonic/articles/455076/>

7. Почему юнит-тесты не работают в научных приложениях. — Режим доступа: <https://habr.com/ru/articles/92038/>

8. Анатомия юнит-теста. — Режим доступа: <https://habr.com/ru/articles/554808/>

9. Автотесты в World of Tanks: боты на страже качества. — Режим доступа: <https://www.youtube.com/watch?v=eyJVEmuxFL4>

10. Тестирование десктоп-приложений с использованием нейронных сетей. — Режим доступа: <https://www.youtube.com/watch?v=GNb-C4qYSQw>

11. Fuzzing-тестирование: ищем баги в JIT-компиляторе и не только. — Режим доступа: https://www.youtube.com/watch?v=S2vBz_ZPAbc

12. Postman Beginner's Course - API Testing. — Режим доступа: <https://www.youtube.com/watch?v=VywxIQ2ZXw4>

13. Использование диаграммы вариантов использования UML при проектировании программного обеспечения. — Режим доступа: <https://habr.com/ru/articles/566218/>

14. UML Use Case Diagrams. — Режим доступа: <https://www.uml-diagrams.org/use-case-diagrams.html>

15. Топ-10 ошибок, найденных PVS-Studio в проектах на ASP.NET Core. — Режим доступа: <https://habr.com/ru/companies/pvs-studio/articles/654395/>

16. Cppcheck is a static analysis tool for C/C++ code. — Режим доступа: <http://cppcheck.sourceforge.net>

17. Valgrind is an instrumentation framework for building dynamic analysis tools. — Режим доступа: <http://www.valgrind.org>

8. Фонд оценочных материалов для проведения текущего контроля успеваемости и промежуточной аттестации

Содержание промежуточной аттестации раскрывается в комплекте контролирующих материалов, предназначенных для проверки соответствия уровня подготовки по дисциплине требованиям ФГОС, которые хранятся на кафедре-разработчике РПД в печатном виде и в ЭИОС.

Фонд оценочных материалов (ФОМ) по дисциплине представлен в приложении А.

9. Перечень информационных технологий, используемых при осуществлении образовательного процесса по дисциплине, включая перечень программного обеспечения и информационных справочных систем

Для успешного освоения дисциплины используются ресурсы электронной информационно-образовательной среды, образовательные интернет-порталы, глобальная компьютерная сеть Интернет. В процессе изучения дисциплины происходит интерактивное взаимодействие обучающегося с преподавателем через личный кабинет студента.

№пп	Используемое программное обеспечение
1	Dev-C++
2	Dia
3	Eclipse IDE
4	LibreOffice
5	Selenium IDE
6	Selenium WebDriver
7	Windows
8	Антивирус Kaspersky

№пп	Используемые профессиональные базы данных и информационные справочные системы
1	IEEE Xplore - Интернет библиотека с доступом к реферативным и полнотекстовым статьям и материалам конференций. Бессрочно без подписки (https://ieeexplore.ieee.org/Xplore/home.jsp)
2	Бесплатная электронная библиотека онлайн "Единое окно к образовательным ресурсам" для студентов и преподавателей; каталог ссылок на образовательные интернет-ресурсы (http://Window.edu.ru)
3	Национальная электронная библиотека (НЭБ) — свободный доступ читателей к фондам российских библиотек. Содержит коллекции оцифрованных документов (как открытого доступа, так и ограниченных авторским правом), а также каталог

№пп	Используемые профессиональные базы данных и информационные справочные системы
	изданий, хранящихся в библиотеках России. (http://нэб.рф/)

10. Описание материально-технической базы, необходимой для осуществления образовательного процесса по дисциплине

Наименование специальных помещений и помещений для самостоятельной работы
учебные аудитории для проведения учебных занятий
помещения для самостоятельной работы

Материально-техническое обеспечение и организация образовательного процесса по дисциплине для инвалидов и лиц с ограниченными возможностями здоровья осуществляется в соответствии с «Положением об обучении инвалидов и лиц с ограниченными возможностями здоровья».