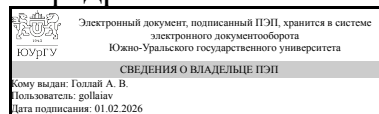


ЮЖНО-УРАЛЬСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ

УТВЕРЖДАЮ:
Заведующий выпускающей
кафедрой



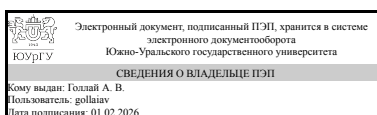
А. В. Голлай

РАБОЧАЯ ПРОГРАММА

дисциплины 1.Ф.ПО.14 Тестирование ПО
для направления 09.03.01 Информатика и вычислительная техника
уровень Бакалавриат
профиль подготовки ИТ-инженерия
форма обучения очная
кафедра-разработчик Центр подготовки топ-специалистов в сфере ИТ "Цифровой
Урал"

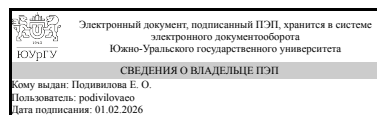
Рабочая программа составлена в соответствии с ФГОС ВО по направлению
подготовки 09.03.01 Информатика и вычислительная техника, утверждённым
приказом Минобрнауки от 19.09.2017 № 929

Зав.кафедрой разработчика,
д.техн.н., доц.



А. В. Голлай

Разработчик программы,
к.техн.н., доцент



Е. О. Подивилова

1. Цели и задачи дисциплины

Цель освоения дисциплины: сформировать у обучающихся системные знания и практические умения в области процессов, методов и инструментов тестирования программного обеспечения, обеспечивающие готовность к участию в обеспечении качества ПО в современных IT-проектах. Задачи освоения дисциплины: дать знания о фундаментальных принципах, видах, уровнях тестирования и его месте в жизненном цикле разработки ПО; сформировать умения применять основные техники тест-дизайна и создавать тестовую документацию (тест-планы, тест-кейсы, чек-листы, баг-репорты); выработать навыки ручного тестирования веб-приложений, включая проверку пользовательского интерфейса, бизнес-логики и данных; познакомить с основами автоматизации тестирования (UI и API) с использованием современных инструментов и фреймворков; обучить работе в среде управления задачами и тестированием и с системами контроля версий.

Краткое содержание дисциплины

Краткое содержание дисциплины охватывает основы тестирования программного обеспечения, включая методы и техники тест-дизайна, виды тестирования (функциональное, нагрузочное, регрессионное и др.), инструменты автоматизации, управление дефектами и процессами тестирования. Рассматриваются принципы построения эффективных тестовых сценариев, методики оценки покрытия кода и метрики качества, а также современные практики DevOps и Agile, направленные на интеграцию тестирования в непрерывный цикл разработки. Дисциплина направлена на подготовку квалифицированных специалистов, способных обеспечить высокое качество выпускаемых программных решений.

2. Компетенции обучающегося, формируемые в результате освоения дисциплины

Планируемые результаты освоения ОП ВО (компетенции)	Планируемые результаты обучения по дисциплине
ПК-2 Способен проверять работоспособность и проводить рефакторинг кода программного обеспечения	Знает: методы и средства проверки работоспособности компьютерного программного обеспечения; государственные стандарты испытания автоматизированных систем; руководящие документы по стандартизации требований к документам автоматизированных систем; методы и средства рефакторинга и инспекции программного кода; нормативно-технические документы (стандарты и регламенты), регламентирующие требования к программному коду, порядок отражения изменений в системе управления версиями, порядок отражения результатов рефакторинга, оптимизации и инспекции в коллективной базе знаний; знает о возможности отказа частей системы в результате воздействий внутреннего и внешнего нарушителя (хакер, неосторожный пользователь, программист, поставщик компонентов) Умеет: применять методы и средства проверки

	работоспособности компьютерного программного обеспечения; интерпретировать диагностические данные проверки работоспособности компьютерного программного обеспечения; анализировать значения полученных характеристик компьютерного программного обеспечения; анализировать программный код на соответствие требованиям по читаемости и производительности; проводить инспекцию программного кода для поиска не обнаруженных на ранних стадиях разработки компьютерного программного обеспечения ошибок и критических мест; применять методы и средства рефакторинга и инспекции программного кода; публиковать результаты рефакторинга и инспекции в коллективной базе знаний; проводить инспекцию программного кода для поиска мест возможных отказов из воздействий внутреннего или внешнего нарушителя Имеет практический опыт: проверки работоспособности компьютерного программного обеспечения с учетом актуальных стандартов и регламентов; применения методов и средств рефакторинга и инспекции программного кода
--	--

3. Место дисциплины в структуре ОП ВО

Перечень предшествующих дисциплин, видов работ учебного плана	Перечень последующих дисциплин, видов работ
Объектно-ориентированное программирование	Не предусмотрены

Требования к «входным» знаниям, умениям, навыкам студента, необходимым при освоении данной дисциплины и приобретенным в результате освоения предшествующих дисциплин:

Дисциплина	Требования
Объектно-ориентированное программирование	Знает: основные возможности современных интегрированных сред разработки программного обеспечения на объектно-ориентированных языках программирования, возможности компиляторов программных проектов под различные операционные системы, наборы инструкций для системных утилит автоматической сборки программного обеспечения и установки программных пакетов объектно-ориентированных библиотек и фреймворков, принципы чистого кода, SOLID, DRY, KISS и др., принципы предметно-ориентированного проектирования (ПОП) программного обеспечения, паттерны проектирования и антипаттерны, методы и средства проверки работоспособности

	компьютерного программного обеспечения, синтаксис выбранного языка программирования, особенности программирования на этом языке, стандартные библиотеки языка программирования; методологии разработки компьютерного программного обеспечения; технологии программирования; возможности используемой системы управления версиями и вспомогательных инструментальных программных средств; установленный регламент использования системы управления версиями Умеет: использовать функциональные возможности современных интегрированных сред разработки программного обеспечения на объектно-ориентированных языках программирования для разработки прикладных программ, использовать утилиты автоматической сборки и развертывания программ в операционных системах, разрабатывать модульный и тестируемый программный код, выполнять модульное, интеграционное и нагрузочное тестирование, проводить рефакторинг для повышения качества кода, применять принципы ПОП при разработке программного обеспечения на языках программирования высокого уровня абстракций и в LowCode и NoCode системах, применять методы и средства проверки работоспособности компьютерного программного обеспечения, применять выбранные языки программирования для написания программного кода; использовать выбранную среду программирования; использовать возможности имеющейся технической и/или программной архитектуры для написания программного кода; регистрировать изменения исходного текста программного кода в системе управления версиями; сохранять изменения программного кода в соответствии с регламентом управления версиями; выполнять слияние, разделение и сравнение исходных текстов программного кода Имеет практический опыт: работы с основными современными интегрированными средами разработки программного обеспечения на объектно-ориентированных языках, разработки, отладки и развёртывания программного обеспечения в операционных системах семейства Windows и Linux
--	---

4. Объём и виды учебной работы

Общая трудоемкость дисциплины составляет 3 з.е., 108 ч., 72,5 ч. контактной работы

Вид учебной работы	Всего часов	Распределение по семестрам в часах
		Номер семестра
		7
Общая трудоёмкость дисциплины	108	108
<i>Аудиторные занятия:</i>	64	64
Лекции (Л)	16	16
Практические занятия, семинары и (или) другие виды аудиторных занятий (ПЗ)	48	48
Лабораторные работы (ЛР)	0	0
<i>Самостоятельная работа (СРС)</i>	35,5	35,5
Выполнение контрольно-рейтинговых мероприятий	35,5	35,5
Консультации и промежуточная аттестация	8,5	8,5
Вид контроля (зачет, диф.зачет, экзамен)	-	экзамен

5. Содержание дисциплины

№ раздела	Наименование разделов дисциплины	Объем аудиторных занятий по видам в часах			
		Всего	Л	ПЗ	ЛР
1	Основы тестирования и методологии	16	4	12	0
2	Автоматизированное и нагрузочное тестирование	16	4	12	0
3	Техники тестирования и работа с ошибками	16	4	12	0
4	Совершенствование процессов и внедрение новых технологий тестирования	16	4	12	0

5.1. Лекции

№ лекции	№ раздела	Наименование или краткое содержание лекционного занятия	Кол-во часов
1	1	Введение в тестирование ПО. Определение и цели тестирования. Роль тестирования в жизненном цикле разработки ПО. Классификация видов тестирования.	2
2	1	Методологии и процессы тестирования Модели жизненного цикла тестирования. Процессы управления качеством. Этапы тестирования: планирование, дизайн, выполнение, отчетность.	2
3	2	Автоматизация тестирования. Инструменты автоматизированного тестирования. Архитектуры фреймворков автоматизации. Практики написания надежных автотестов.	2
4	2	Нагрузочное и стресс-тестирование. Цели и особенности нагрузочного тестирования. Средства моделирования нагрузки. Анализ производительности и масштабируемости системы.	2
5	3	Функциональное и нефункциональное тестирование Функциональное тестирование: black-box, white-box, grey-box. Нефункциональное тестирование: производительность, безопасность, совместимость. Типичные дефекты и сценарии проверки.	2
6	3	Регрессионное и приемочное тестирование. Понятие регрессии и необходимость регрессионного тестирования. Стратегии выбора регрессионных тестов. Приемочные тесты.	2
7	4	Управление дефектами и тестирование в условиях Agile. Жизненный цикл дефекта: обнаружение, фиксация, исправление, закрытие. Использование баг-	2

		трекинга. Особенности организации тестирования в гибких методологиях (Scrum, Kanban).	
8	4	Современные тенденции и перспективы тестирования. Обзор современных подходов (DevOps, Continuous Integration/Continuous Delivery). Тенденции развития профессии QA Engineer. Будущие направления исследований и инновационные технологии тестирования.	2

5.2. Практические занятия, семинары

№ занятия	№ раздела	Наименование или краткое содержание практического занятия, семинара	Кол-во часов
1	1	Изучение документации проекта. Анализ документации проекта (техническое задание, спецификации). Определение требований, подлежащих проверке.	2
2	1	Планирование тестирования	2
3	1	Разработка тест-кейсов Разработка комплекта функциональных тест-кейсов для заданного функционала приложения.	2
4	1	Выполнение ручного функционального тестирования	2
5	1	Анализ покрытия кода	2
6	1	Настройка среды автоматического тестирования	2
7	2	Создание первых автоматических тестов.	2
8	2	Отладка и оптимизация автоматических тестов	2
9	2	Проектирование нагрузочных тестов	2
10	2	Запуск нагрузочного теста	2
11	2	Анализ рисков и критичности дефектов	2
12	2	Регрессия и обратная проверка изменений	2
13	3	Безопасностное тестирование	2
14	3	Непрерывная интеграция и доставка	2
15	3	Тестирование мобильных приложений	2
16	3	Документация дефектов	2
17	3	Оценка качества тестирования	2
18	3	Составление отчёта о тестировании	2
19	4	Имитация отказоустойчивых ситуаций	2
20	4	Улучшение юзабилити интерфейса	2
21	4	Работа с системой контроля версий Git	2
22	4	Повышение эффективности тестирования Применение техники Data Driven Testing (DDT) для ускорения процесса тестирования.	2
23	4	Оценка зрелости процесса тестирования	2
24	4	Комплексное тестирование	2

5.3. Лабораторные работы

Не предусмотрены

5.4. Самостоятельная работа студента

Выполнение СРС			
Подвид СРС	Список литературы (с указанием разделов, глав, страниц) / ссылка на ресурс	Семестр	Кол-во часов

Выполнение контрольно-рейтинговых мероприятий	Аниче, М. Эффективное тестирование программного обеспечения / М. Аниче ; перевод с английского А. Н. Киселева. — Москва : ДМК Пресс, 2023. — 370 с. — ISBN 978-5-97060-997-2. — Текст : электронный // Лань : электронно-библиотечная система.	7	35,5
---	--	---	------

6. Фонд оценочных средств для проведения текущего контроля успеваемости, промежуточной аттестации

Контроль качества освоения образовательной программы осуществляется в соответствии с Положением о балльно-рейтинговой системе оценивания результатов учебной деятельности обучающихся.

6.1. Контрольные мероприятия (КМ)

№ КМ	Семестр	Вид контроля	Название контрольного мероприятия	Вес	Макс. балл	Порядок начисления баллов	Учитывается в ПА
1	7	Текущий контроль	Аудит требований и проектирование тестового покрытия	1	5	1 балл - задание выполнено не полностью, не продемонстрированы результаты выполнения практической работы 2 балла - задание выполнено не полностью, не даны ответы на теоретические вопросы 3 балла - продемонстрированы результаты выполнения практической работы, есть существенные ошибки в решении, даны не четкие ответы на теоретические вопросы 4 балла - продемонстрированы результаты выполнения практической работы, есть незначительные ошибки в решении или даны не четкие ответы на теоретические вопросы 5 баллов - продемонстрированы результаты выполнения практической работы, нет ошибок в результатах задания, даны ответы на теоретические вопросы по материалам данной работы	экзамен
2	7	Текущий контроль	Эвристический аудит и юзабилити-тестирование	1	5	1 балл - задание выполнено не полностью, не продемонстрированы результаты выполнения практической работы 2 балла - задание выполнено не полностью, не даны ответы на теоретические вопросы 3 балла - продемонстрированы результаты выполнения практической работы, есть существенные ошибки в решении, даны не четкие ответы на теоретические вопросы	экзамен

						4 балла - продемонстрированы результаты выполнения практической работы, есть незначительные ошибки в решении или даны не четкие ответы на теоретические вопросы 5 баллов - продемонстрированы результаты выполнения практической работы, нет ошибок в результатах задания, даны ответы на теоретические вопросы по материалам данной работы	
3	7	Текущий контроль	Автоматизация тестирования	1	5	1 балл - задание выполнено не полностью, не продемонстрированы результаты выполнения практической работы 2 балла - задание выполнено не полностью, не даны ответы на теоретические вопросы 3 балла - продемонстрированы результаты выполнения практической работы, есть существенные ошибки в решении, даны не четкие ответы на теоретические вопросы 4 балла - продемонстрированы результаты выполнения практической работы, есть незначительные ошибки в решении или даны не четкие ответы на теоретические вопросы 5 баллов - продемонстрированы результаты выполнения практической работы, нет ошибок в результатах задания, даны ответы на теоретические вопросы по материалам данной работы	экзамен
4	7	Текущий контроль	Тестирование API	1	5	1 балл - задание выполнено не полностью, не продемонстрированы результаты выполнения практической работы 2 балла - задание выполнено не полностью, не даны ответы на теоретические вопросы 3 балла - продемонстрированы результаты выполнения практической работы, есть существенные ошибки в решении, даны не четкие ответы на теоретические вопросы 4 балла - продемонстрированы результаты выполнения практической работы, есть незначительные ошибки в решении или даны не четкие ответы на теоретические вопросы 5 баллов - продемонстрированы результаты выполнения практической работы, нет ошибок в результатах задания, даны ответы на теоретические вопросы по материалам данной работы	экзамен
5	7	Промежуточная	Экзамен	-	20	Экзамен в форме тестирования из 20 вопросов. За каждый правильный ответ	экзамен

		аттестация			- 1 балл	
--	--	------------	--	--	----------	--

6.2. Процедура проведения, критерии оценивания

Не предусмотрены

6.3. Паспорт фонда оценочных средств

Компетенции	Результаты обучения	№ КМ				
		1	2	3	4	5
ПК-2	Знает: методы и средства проверки работоспособности компьютерного программного обеспечения; государственные стандарты испытания автоматизированных систем; руководящие документы по стандартизации требований к документам автоматизированных систем; методы и средства рефакторинга и инспекции программного кода; нормативно-технические документы (стандарты и регламенты), регламентирующие требования к программному коду, порядок отражения изменений в системе управления версиями, порядок отражения результатов рефакторинга, оптимизации и инспекции в коллективной базе знаний; знает о возможности отказа частей системы в результате воздействий внутреннего и внешнего нарушителя (хакер, неосторожный пользователь, программист, поставщик компонентов)		++			+
ПК-2	Умеет: применять методы и средства проверки работоспособности компьютерного программного обеспечения; интерпретировать диагностические данные проверки работоспособности компьютерного программного обеспечения; анализировать значения полученных характеристик компьютерного программного обеспечения; анализировать программный код на соответствие требованиям по читаемости и производительности; проводить инспекцию программного кода для поиска не обнаруженных на ранних стадиях разработки компьютерного программного обеспечения ошибок и критических мест; применять методы и средства рефакторинга и инспекции программного кода; публиковать результаты рефакторинга и инспекции в коллективной базе знаний; проводить инспекцию программного кода для поиска мест возможных отказов из воздействий внутреннего или внешнего нарушителя		++	++	++	++
ПК-2	Имеет практический опыт: проверки работоспособности компьютерного программного обеспечения с учетом актуальных стандартов и регламентов; применения методов и средств рефакторинга и инспекции программного кода			++		

Типовые контрольные задания по каждому мероприятию находятся в приложениях.

7. Учебно-методическое и информационное обеспечение дисциплины

Печатная учебно-методическая документация

а) основная литература:

Не предусмотрена

б) дополнительная литература:

Не предусмотрена

в) отечественные и зарубежные журналы по дисциплине, имеющиеся в библиотеке:

Не предусмотрены

г) методические указания для студентов по освоению дисциплины:

1. Установка и настройка Selenium

из них: учебно-методическое обеспечение самостоятельной работы студента:

1. Установка и настройка Selenium

Электронная учебно-методическая документация

№	Вид литературы	Наименование ресурса в электронной форме	Библиографическое описание
1	Основная литература	ЭБС издательства Лань	Аниче, М. Эффективное тестирование программного обеспечения / М. Аниче ; перевод с английского А. Н. Киселева. — Москва : ДМК Пресс, 2023. — 370 с. — ISBN 978-5-97060-997-2. — Текст : электронный // Лань : электронно-библиотечная система. https://e.lanbook.com/book/314930
2	Дополнительная литература	ЭБС издательства Лань	Зубкова, Т. М. Технология разработки программного обеспечения : учебное пособие / Т. М. Зубкова. — Санкт-Петербург : Лань, 2022. — 324 с. — ISBN 978-5-8114-3842-6. — Текст : электронный // Лань : электронно-библиотечная система. https://e.lanbook.com/book/206882
3	Основная литература	ЭБС издательства Лань	Бубнов, А. А. Тестирование программного обеспечения : учебное пособие / А. А. Бубнов, С. А. Бубнов, В. В. Тишкина. — Рязань : РГРТУ, 2024. — 164 с. — ISBN 978-5-7722-0421-4. — Текст : электронный // Лань : электронно-библиотечная система. https://e.lanbook.com/book/494540
4	Основная литература	ЭБС IPR SMART	Методы тестирования и отладки программного обеспечения: учебник / Е.Е. Карпович. – М. : Изд. Дом НИТУ «МИСиС», 2020. – 136 с. https://www.iprbookshop.ru/epd-reader?publicationId=106722

Перечень используемого программного обеспечения:

1. Microsoft-Visual Studio(бессрочно)

Перечень используемых профессиональных баз данных и информационных справочных систем:

Нет

8. Материально-техническое обеспечение дисциплины

Вид занятий	№ ауд.	Основное оборудование, стенды, макеты, компьютерная техника, предустановленное программное обеспечение, используемое для различных видов занятий
Лекции		Мультимедийная аудитория с проектором
Практические занятия и семинары		Компьютерный класс