

Документ подписан простой электронной подписью  
Информация о владельце:  
ФИО: Галунин Сергей Александрович  
Должность: проректор по учебной работе  
Дата подписания: 23.12.2025 12:07:09  
Уникальный программный ключ:  
08ef34338325bdb0ac5a47baa5472ce36cc3fc3b

Приложение к ОПОП  
«Информационно-управляющие  
системы»



**СПбГЭТУ «ЛЭТИ»**  
ПЕРВЫЙ ЭЛЕКТРОТЕХНИЧЕСКИЙ

МИНОБРНАУКИ РОССИИ

федеральное государственное автономное образовательное учреждение высшего образования  
**«Санкт-Петербургский государственный электротехнический университет  
«ЛЭТИ» им. В.И.Ульянова (Ленина)»  
(СПбГЭТУ «ЛЭТИ»)**

---

**РАБОЧАЯ ПРОГРАММА**

дисциплины

**«АВТОМАТИЗАЦИЯ ТЕСТИРОВАНИЯ»**

для подготовки бакалавров

по направлению

09.03.02 «Информационные системы и технологии»

по профилю

**«Информационно-управляющие системы»**

Санкт-Петербург

2025

## **ЛИСТ СОГЛАСОВАНИЯ**

Разработчики:

доцент, к.т.н. Турнецкая Е.Л.

Рабочая программа рассмотрена и одобрена на заседании кафедры ИС  
16.01.2025, протокол № 1

Рабочая программа рассмотрена и одобрена учебно-методической комиссией  
ФКТИ, 28.01.2025, протокол № 1

Согласовано в ИС ИОТ

Начальник ОМОЛА Загороднюк О.В.

## **1 СТРУКТУРА ДИСЦИПЛИНЫ**

Обеспечивающий факультет	ФКТИ
Обеспечивающая кафедра	ИС
Общая трудоемкость (ЗЕТ)	3
Курс	3
Семестр	5

## Виды занятий

Электронные лекции (акад. часов)	34
Электронные практические (академ. часов) (академ. часов)	34
Иная контактная работа (академ. часов)	1
Все контактные часы (академ. часов)	1
Самостоятельная работа, включая часы на контроль (академ. часов)	39
Всего (академ. часов)	108

## **Вид промежуточной аттестации**

Дифф. зачет (курс) 3

## **2 АННОТАЦИЯ ДИСЦИПЛИНЫ**

### **«АВТОМАТИЗАЦИЯ ТЕСТИРОВАНИЯ»**

Содержание дисциплины охватывает круг вопросов, связанных с автоматизацией процесса тестирования, применением фреймворков и библиотек для программной реализации тестов. Рассмотрены вопросы автоматизации тестирования веб-приложений с использованием программного стека на основе языка программирования Python, фреймворка PyTest и библиотеки Selenium WebDriver. Представлены варианты ручного и автоматизированного поиска элементов на веб-странице с помощью CSS-селекторов и XPath-путей. Показаны возможности составления структуры автотеста средствами браузерного расширения Selenium IDE.

#### **SUBJECT SUMMARY**

#### **«TEST AUTOMATION»**

The content of the discipline covers a range of issues related to the automation of the testing process, the use of frameworks and libraries for software implementation of tests. The issues of automation of web application testing using a software stack based on the Python programming language, the PyTest framework and the Selenium WebDriver library are considered. Options for manual and automated search for elements on a web page using CSS selectors and XPath paths are presented. The possibilities of compiling an autotest structure using the Selenium IDE browser extension are shown.

## **3 ОБЩИЕ ПОЛОЖЕНИЯ**

### **3.1 Цели и задачи дисциплины**

1. Цели изучения дисциплины: приобретение системных знаний и навыков тестирования веб-приложений с помощью специализированных программных инструментов, а также автоматизированного контроля качества программного обеспечения с помощью специализированных фреймворков и библиотек языка программирования Python
2. Поставленная цель реализуется в процессе решения следующих задач:
  - приобретение системных знаний в области автоматизации тестирования;
  - изучение фреймворков автоматизации тестирования;
  - получение практических навыков реализации тестовых мероприятий при использовании специализированных фреймворков и библиотек языка программирования Python
3. Студент приобретает знания о(б):
  - описании тестовых случаев для выполнения процесса тестирования программного обеспечения, включая разработку автоматизированных тестов;
  - фреймворках и библиотеках для проведения автоматизированных тестовых мероприятий
4. У студента формируются умения:
  - составлять сценарий автоматизированного тестирования ПО по разработанным тестовым случаям
  - проводить автоматизированное тестирование клиентской части программной системы;
  - выбирать и использовать инструменты тестирования для решения конкретной задачи.
5. В ходе изучения дисциплины студент овладевает навыками автоматизиро-

ванного тестирования клиентской части веб-приложений с применением фреймворков.

### **3.2 Место дисциплины в структуре ОПОП**

Дисциплина изучается на основе ранее освоенных дисциплин учебного плана:

1. «Основы тестирования программного обеспечения»
2. «Тестирование программного обеспечения»

и обеспечивает изучение последующих дисциплин:

1. «Промышленное тестирование программного обеспечения»
2. «Проектирование графического пользовательского интерфейса»
3. «Инженерный документооборот»
4. «Качество и метрология программного обеспечения»

### **3.3 Перечень планируемых результатов обучения по дисциплине, соотнесенных с планируемыми результатами освоения образовательной программы**

В результате освоения образовательной программы обучающийся должен достичь следующие результаты обучения по дисциплине:

<b>Код компетенции/ индикатора компетенции</b>	<b>Наименование компетенции/индикатора компетенции</b>
ПК-0	Способен разрабатывать информационные модели и применять их для решения задач профессиональной деятельности
ПК-0.1	<i>Знает современные виды информационных моделей, применяемых при решении задач профессиональной деятельности</i>
ПК-0.2	<i>Создает и модифицирует информационные модели для решения задач профессиональной деятельности</i>
ПК-0.3	<i>Применяет информационные модели для решения задач профессиональной деятельности</i>

## **4 СОДЕРЖАНИЕ ДИСЦИПЛИНЫ**

### **4.1 Содержание разделов дисциплины**

#### **4.1.1 Наименование тем и часы на все виды нагрузки**

<b>№ п/п</b>	<b>Наименование темы дисциплины</b>	<b>ЭЛек, ач</b>	<b>ЭПр, ач</b>	<b>ИКР, ач</b>	<b>СР, ач</b>
1	Тестирование клиентской части веб-приложений на основе локаторов	6	10	1	5
2	Автоматизация тестирования	4			5
3	Элементы автоматизации веб-приложений	4	6		5
4	Инструмент управления браузером Selenium WebDriver	6	6		5
5	Фреймворки Python для тестирования	4			5
6	Тестовый фреймворк PyTest	6	6		5
7	Паттерн Page Object	4	6		9
	Итого, ач	34	34	1	39
	Из них ач на контроль	0	0	0	0
	Общая трудоемкость освоения, ач/зе				108/3

#### **4.1.2 Содержание**

<b>№ п/п</b>	<b>Наименование темы дисциплины</b>	<b>Содержание</b>
1	Тестирование клиентской части веб-приложений на основе локаторов	Объектная модель документа DOM-модель. Элементы автоматизации web-приложений. Поиск элементов на web-странице. Локаторы и селекторы в Selenium. CSS-локаторы. XPath-пути. Инструменты web-разработчика в браузере.
2	Автоматизация тестирования	Подходы к автоматизации. Преимущества и недостатки автоматизации. Программные инструменты проекта Selenium для автоматизации тестирования
3	Элементы автоматизации веб-приложений	Инструмент Selenium IDE. Принцип работы. Структура теста. Проверка наличия элементов. Проверка атрибутов и текста. Запись сценария. Сохранение сценария.
4	Инструмент управления браузером Selenium WebDriver	Библиотека Selenium WebDriver. Установка виртуального окружения проекта. Практическое применение Selenium WebDriver. Методы реализации ожиданий в Selenium.
5	Фреймворки Python для тестирования	Принципы ООП при разработке клиентской части web-приложений. Структура фреймворка. Фреймворки Python для тестирования. Фреймворк Behave. Фреймворки Unittest и PyTest.

<b>№ п/п</b>	<b>Наименование темы дисциплины</b>	<b>Содержание</b>
6	Тестовый фреймворк PyTest	PyTest: преимущества и недостатки. Правила запуска тестов, отчёты. Фикстуры и маркеры в Pytest. Тестирование с помощью PyTest и Selenium.
7	Паттерн Page Object	Архитектура шаблона Page Object..Реализация тестирования на основе паттерна Page Object на Python с использованием библиотек Selenium и Pytest.

#### **4.2 Перечень лабораторных работ**

Лабораторные работы не предусмотрены.

#### **4.3 Перечень практических занятий**

<b>Наименование практических занятий</b>	<b>Количество ауд. часов</b>
1. Поиск CSS-локаторов средствами DevTools	6
2. Поиск XPaht-локаторов средствами DevTools	4
3. Разработка сценария автоматизированного тестирования для Selenium	6
4. Тестирование с помощью Webdriwer и Python	6
5. Тестирование на основе PyTest	6
6. Тестирование на основе паттерна Page Object	6
Итого	34

#### **4.4 Курсовое проектирование**

Курсовая работа (проект) не предусмотрены.

#### **4.5 Реферат**

Реферат не предусмотрен.

#### **4.6 Индивидуальное домашнее задание**

Индивидуальное домашнее задание не предусмотрено.

#### **4.7 Доклад**

Доклад не предусмотрен.

#### **4.8 Кейс**

Кейс не предусмотрен.

#### **4.9 Организация и учебно-методическое обеспечение самостоятельной работы**

Дисциплина реализуется в формате онлайн-курса. Учебно-методическое сопровождение самостоятельной работы включает материалы онлайн-курса, состоящие из видеолекций, конспекта к каждой лекции и методических указаний по выполнению практических заданий. Перед выполнением самостоятельной работы студентам рекомендуется ознакомиться с имеющимися информационными и учебно-методическими материалами. Также самостоятельная работа включает знакомство с рекомендованными в курсе информационными ресурсами сети Интернет и специализированными бесплатными онлайн-сервисами.

Планирование времени для изучения дисциплины осуществляется на весь период обучения, предусматривая при этом регулярное повторение пройденного материала. Обучающимся, в рамках внеаудиторной самостоятельной работы, необходимо регулярно дополнять сведениями из литературных источников материал, законспектированный на лекциях. При этом на основе изучения рекомендованной литературы целесообразно составить конспект основных положений, терминов и определений, необходимых для освоения разделов учебной дисциплины.

Особое место уделяется консультированию, как одной из форм обучения и контроля самостоятельной работы. Консультирование предполагает особым образом организованное взаимодействие между преподавателем и студентами, при этом предполагается, что консультант либо знает готовое решение, которое он может предписать консультируемому, либо он владеет способами деятельности, которые указывают путь решения проблемы.

<b>Текущая СРС</b>	<b>Примерная трудоемкость, ач</b>
Работа с лекционным материалом, с учебной литературой	4
Опережающая самостоятельная работа (изучение нового материала до его изложения на занятиях)	0
Самостоятельное изучение разделов дисциплины	6
Выполнение домашних заданий, домашних контрольных работ	0
Подготовка к лабораторным работам, к практическим и семинарским занятиям	6
Подготовка к контрольным работам, коллоквиумам	6
Выполнение расчетно-графических работ	0
Выполнение курсового проекта или курсовой работы	0
Поиск, изучение и презентация информации по заданной проблеме, анализ научных публикаций по заданной теме	0
Работа над междисциплинарным проектом	0
Анализ данных по заданной теме, выполнение расчетов, составление схем и моделей, на основе собранных данных	0
Подготовка к зачету, дифференцированному зачету, экзамену	17
<b>ИТОГО СРС</b>	<b>39</b>

## **5 Учебно-методическое обеспечение дисциплины**

### **5.1 Перечень основной и дополнительной литературы, необходимой для освоения дисциплины**

<b>№ п/п</b>	<b>Название, библиографическое описание</b>	<b>К-во экз. в библ.</b>
<b>Основная литература</b>		
1	Чернышев, Станислав Андреевич. Основы программирования на Python : учебное пособие для вузов / С. А. Чернышев., 2024. -349 с. -Текст : электронный.	неогр.
2	Персиваль Г. Python. Разработка на основе тестирования. Повинуйся Билли-тестировщику, используя Django, Selenium и JavaScript / Г. Персиваль, 2018. -622 с. -Текст : электронный.	неогр.
3	Игнатьев А. В. Тестирование программного обеспечения : учебное пособие для вузов / А. В. Игнатьев, 2023. -56 с. -Текст : электронный.	неогр.
4	Аниче М. Эффективное тестирование программного обеспечения / М. Аниче, 2023. -370 с. -Текст : непосредственный.	неогр.
<b>Дополнительная литература</b>		
1	Старолетов С. М. Основы тестирования и верификации программного обеспечения : учебное пособие для вузов / С. М. Старолетов, 2023. -344 с. -Текст : электронный.	неогр.
2	Митчелл Р. Сcraping веб-сайтов с помощью Python : руководство / Р. Митчелл, 2016. -280 с. -Текст : электронный.	неогр.
3	Ленц М. Python Непрерывная интеграция и доставка : руководство / М. Ленц, 2020. -168 с. -Текст : электронный.	неогр.
4	Борзунов С. В. Языки программирования. Python: решение сложных задач : учебное пособие для вузов / С. В. Борзунов, С. Д. Кургалин, 2023. -192 с. -Текст : электронный.	неогр.

### **5.2 Перечень ресурсов информационно-телекоммуникационной сети «Интернет», используемых при освоении дисциплины**

<b>№ п/п</b>	<b>Электронный адрес</b>
1	Python: основы и применение <a href="https://stepik.org/course/512">https://stepik.org/course/512</a>
2	Вселенная тестирования, или Как стать тестировщиком <a href="https://stepik.org/course/118842">https://stepik.org/course/118842</a>
3	Автоматизация тестирования с помощью Selenium и Python <a href="https://stepik.org/course/575">https://stepik.org/course/575</a>
4	Пакета программного обеспечения с открытым исходным кодом Selenium <a href="https://www.selenium.dev/">https://www.selenium.dev/</a>

№ п/п	Электронный адрес
5	Документация по PyTest <a href="https://pytest-docs-ru.readthedocs.io/ru/latest/contents.html">https://pytest-docs-ru.readthedocs.io/ru/latest/contents.html</a>
6	”Selenium на Python” от Алексея Коледачкина <a href="https://stepik.org/course/188355/">https://stepik.org/course/188355/</a>

### 5.3 Адрес сайта курса

Адрес сайта курса: [https://open.etu.ru/courses/course-v1:etu+AST+fall\\_2024/course/](https://open.etu.ru/courses/course-v1:etu+AST+fall_2024/course/)

## **6 Критерии оценивания и оценочные материалы**

### **6.1 Критерии оценивания**

Для дисциплины «Автоматизация тестирования» формой промежуточной аттестации является дифф. зачет. Оценивание качества освоения дисциплины производится с использованием рейтинговой системы.

#### **Дифференцированный зачет**

<b>Оценка</b>	<b>Количество баллов</b>	<b>Описание</b>
Неудовлетворительно	0 – 54	теоретическое содержание курса не освоено, необходимые практически навыки и умения не сформированы, выполненные учебные задания содержат грубые ошибки, дополнительная самостоятельная работа над курсом не приведет к существенному повышению качества выполнения учебных заданий
Удовлетворительно	55 – 69	теоретическое содержание курса освоено частично, но пробелы не носят существенного характера, необходимые практические навыки и умения работы с освоенным материалом в основном сформированы, большинство предусмотренных программой обучения учебных заданий выполнено, некоторые из выполненных заданий содержат ошибки
Хорошо	70 – 84	теоретическое содержание курса освоено полностью, без пробелов, некоторые практические навыки и умения сформированы недостаточно, все предусмотренные программой обучения учебные задания выполнены, качество выполнения ни одного из них не оценено минимальным числом баллов, некоторые виды заданий выполнены с ошибками
Отлично	85 – 100	теоретическое содержание курса освоено полностью, без пробелов, необходимые практические навыки и умения сформированы, все предусмотренные программой обучения учебные задания выполнены, качество их выполнения оценено количеством баллов, близким к максимальному

## **Особенности допуска**

Студент, выполнивший все точки контроля в течение семестра допускается к дифференциальному зачету. Зачет с оценкой выставляется по результатам текущего контроля в соответствии с рейтинговой системой. Для допуска к дифференцированному зачету студенту следует набрать не менее 55 рейтинговых баллов, которые могут быть набраны при прохождении контрольных точек курса.

## **6.2 Оценочные материалы для проведения текущего контроля и промежуточной аттестации обучающихся по дисциплине**

### **Вопросы к дифф.зачету**

<b>№ п/п</b>	<b>Описание</b>
1	Поиск CSS-локаторов средствами DevTools. Приведите примеры
2	Поиск XRaht-локаторов средствами DevTools. Приведите примеры
3	Преимущества и недостатки автоматизации тестирования
4	Элементы автоматизации web-приложений. Приведите примеры
5	Назначение и возможности инструмента управления браузером Selenium WebDriver
6	Назначение программных продуктов семейства Selenium
7	Применение браузерного расширения Selenium IDE. Основные команды
8	Методы реализации ожиданий в Selenium
9	Структура фреймворка автоматизации тестирования согласно ISTQB
10	Фреймворк Behave. Назначение и основные принципы тестирования
11	Проведите сравнения между фреймворками Unittest и PyTest
12	Фикстуры и маркеры в Pytest
13	Архитектура шаблона Page Object. Приведите пример
14	Объектная модель документа DOM-модель веб-страницы и элементы автоматизации
15	Объясните, что такое объектная модель веб-документов. Приведите схему
16	Маршруты поиска веб-элементов с использованием XPath
17	Применение технологий «Запись и воспроизведение» и «Тестирование на основе ключевых слов» в Selenium IDE
18	Этапы реализации сценария автоматизированного тестирования в Selenium IDE
19	Процесс управления браузером при автоматизации тестирования
20	Назначение формата описания тестов Gherkin

## **Вариант теста**

### ***Итоговый тест***

1. Прочитайте текст и выберите один правильный ответ.

Определите назначение тега <img> при реализации веб-страницы

- a. =для внедрения изображений в графическом формате
- б. для добавления гиперссылки на другой элемент
- в. для добавления контента в текстовом формате

2. Прочитайте текст и выберите один правильный ответ.

Определите количество потомков (включая дочерних) у элемента <ul> в примере ниже:

```
<ul>  
    <li>Земля </li>  
    <li>Марс</li>  
    <li>Марс</li>  
</ul>
```

Ответы:

- а. 0
- б. 2
- в. 4
- г. =6

3. Прочитайте текст и выберите один правильный ответ.

Определите количество потомков (включая дочерних) у элемента <ul> в примере ниже:

```
<ul>
```

```
<li>Привет</li>  
<li>Мир</li>  
</ul>
```

Ответы:

а. 0

б. 2

в. =4

г. 6

4. Прочитайте текст и выберите один правильный ответ.

Определите возможный путь (обход) DOM-дерева при поиске элементов с помощью CSS-локаторов

а. = только в направлении сверху вниз

б. в направлении снизу верх

в. в любом вертикальном или горизонтальном направлении

5. Прочитайте текст и выберите один правильный ответ.

Выберите название XPath-пути в запросе: /html/body/div[1]/h1

а. = абсолютный путь

б. относительный путь

6. Прочитайте текст и выберите один правильный ответ.

Выберите название XPath-пути в запросе: //div[@id='main']/h1

а. абсолютный путь

б. =относительный путь

7. Прочитайте текст и выберите один правильный ответ.

Подберите такой XPath-селектор, чтобы выбрать только кнопку с текстом ЛЭТИ. Используйте XPath-запрос с методом text().

- а. `=//button[contains(text(), "ЛЭТИ")]`
- б. `//button[text("ЛЭТИ")]`

8. Прочитайте текст и выберите несколько правильных ответов.

Выберите по каким идентификаторам можно найти элемент на веб-странице

- а. `=id`
- б. `=тегу`
- в. `=значению атрибута`
- г. `=name`
- д. `=class`

9. Прочитайте текст и выберите один правильный ответ.

Определите, что означает XPath–запрос: `//input`

- а. `=Найти узлы от текущего узла`
- б. `Найти родителя текущего узла`
- в. `Выбрать текущий узел`

10. Прочитайте текст и выберите один правильный ответ.

Определите, что означает XPath–запрос: `//@text`

- а. `Найти узлы от текущего узла`
- б. `=Выбрать конкретный атрибут`
- в. `Выбрать текущий узел`

11. Прочитайте текст и выберите один правильный ответ.

Определите, что означает XPath–запрос: `//div[@*];`

- a. =Выбрать любой элемент
- б. Найти родителя текущего узла
- в. Выбрать текущий узел

12. Прочтите текст и выберите один правильный ответ.

Зафиксируйте, слова «предикат» и «подстановочное выражение» являются определениями одного и того же понятия

- a. Да, синонимы, обозначают одно и то же
- б. = Нет, разные понятия

13. Прочтите текст и установите соответствие. К каждой позиции, данной в левом столбце, подберите соответствующую позицию в правом столбце.

Постройте пирамиду тестирования, где 1 уровень соответствует тестам, которые лежат в основании пирамиды; 4 уровень показывает ее вершину.

- = Первый уровень -> юнит-тесты
- = Второй уровень -> интеграционные тесты
- = Третий уровень -> автоматизированные end-to-end тесты
- = Четвертый уровень -> ручные исследовательские тесты

14. Прочтите текст и выберите несколько правильных ответов.

Выберите, на каких технологиях основана работа Selenium IDE

- a. =Запись и воспроизведение
- б. =Тестирование на основе ключевых слов
- в. Тестирование под управлением поведением

15. Прочтите текст и установите соответствие. К каждой позиции, данной в левом столбце, подберите соответствующую позицию в правом столбце.

При записи действий тестировщика Selenium IDE использует специаль-

ный язык, в котором предусмотрено три класса команд. Установите соответствие между их названием и описанием.

- = Команды класса action -> помогают изменить состояние приложений
- = Команды класса accessors -> включают проверку и сохранение состояния приложения, которые могут быть использованы в последующих командах
- = Команды класса assertion -> проверяют соответствие состояния приложения после выполнения тестового примера желаемому состоянию

16. Прочитайте текст и установите соответствие. К каждой позиции, данной в левом столбце, подберите соответствующую позицию в правом столбце.

При тестировании применяют явные и неявные ожидания. Установите соответствие между определением и описанием

- = Явные ожидания (Explicit Waits) -> используют для определенных условий и позволяют дождаться выполнения определенного условия перед продолжением работы.
- = Неявные ожидания (Implicit Waits) -> применяют ко всем операциям и задают максимальное время ожидания для всех элементов.

17. Прочитайте текст и выберите один правильный ответ.

Программные методы библиотеки Selenium Webdriver поддерживают конкретную версию браузера. То есть, для каждой версии браузера следует скачивать Webdriver, в котором заложены методы взаимодействия с этой версией. Определите, является ли данное утверждение верным.

- a. =Да, требуется скачивать Webdriver под конкретную версию браузера
- b. Нет, достаточно скачать один раз и затем настроить программное окружение

18. Прочитайте текст и установите соответствие. К каждой позиции, дан-

ной в левом столбце, подберите соответствующую позицию в правом столбце.

Установите соответствие

= Webdriver -> сущность, ответственная за управление браузером

= WebElement -> сущность, представляющая собой абстракцию над веб-элементом

= By -> абстракция над локатором веб-элемента.

19. Прочитайте текст и установите соответствие. К каждой позиции, данной в левом столбце, подберите соответствующую позицию в правом столбце.

Приведен листинг программного кода по открытию браузера. Установите соответствие между комментариями и их описанием

```
from selenium import webdriver
```

# Комментарий 1

```
browser = webdriver.Firefox()
```

# Комментарий 2

```
browser.get('http://selenium.dev/')
```

а. = Комментарий 1 -> Создание экземпляра WebDriver для браузера Firefox

б. = Комментарий 2 -> Открытие веб-страницы

20. Прочитайте текст и выберите один правильный ответ.

Определите, результат выполнения команды:

```
elem = browser.find_element(By.NAME, 'p')
```

а. = осуществлен поиск элемента по атрибуту Name

б. = осуществлен поиск всех элементов по атрибуту Name

21. Прочитайте текст и выберите один правильный ответ.

Выберите способ описания фикстуры в PyTest

а. =@pytest.fixture

б. pytest.fixture

22. Прочтите текст и выберите один правильный ответ.

Выберите способ добавления маркеров в PyTest

а. В виде декораторов к функциям, например @pytest.mark.xfail

б. В виде функций или классов, например mark.skip(reason='Тестовый пропуск')

## **Образцы задач (заданий) для контрольных (проверочных) работ**

Вопросы в тестах формируются аналогично приведенным ниже примерам:

***Тест 1.***

***Тема 2. Автоматизация тестирования***

1. Прочтайте текст и выберите несколько правильных ответов

Определите преимущества, которые предоставляет автоматизация тестирования при разработке качественного программного обеспечения.

а. =ускорение процесса тестирования

б. =уменьшение сроков тестирования

в. =уменьшение риска ошибок пропуска инцидента

г. =увеличение затрат на тестирование

2. Прочтайте текст и выберите несколько правильных ответов

Выберите программные инструменты, которые применяют для автоматизации тестирования.

а. =Selenium

б. Kaiten

в. =JMeter

г. =Postman

3. Прочитайте текст и выберите правильный ответ.

Определите, какие тесты необходимо автоматизировать в первую очередь.

а. Тесты, которые требуют повторения несколько раз

б. Тесты, не требующие ввода большого объема данных

в. Тесты, которые не требуют быстрого выявления ошибок

4. Прочитайте текст и выберите несколько правильных ответов

Определите, что представляет собой непрерывная интеграция в разработке программного обеспечения?

а. =Регулярное соединение изменений кода с основной веткой проекта разработки и выполнение автоматических тестов

б. =Регулярное соединение изменений кода с основной веткой проекта разработки, но запуск автотестов только перед выпуском новой версии программного продукта

в. Интеграция изменений с основной веткой проекта раз в месяц с одновременным запуском автотестов

5. Прочитайте текст и выберите несколько правильных ответов

Определите характеристики системы управления проектами, которые необходимо учитывать при ее выборе.

а. =Интеграция с автоматическими тестами

б. =Создание автоматизированных отчетов о тестировании,

в. =Наличие встроенной баг-трекинговой системы

г. Наличие встроенной графической библиотеки

6. Прочитайте текст и выберите правильный ответ.

Выберите отечественную систему управления тестированием.

а. TestNG.

б. = Test It.

в. Selenium

г. Jira

7. Прочитайте текст и выберите несколько правильных ответов

Зафиксируйте фреймворки на основе Python, которые используют для автоматизации тестов?

а. JUnit

б. = Robot Framework

в. TestNG

г. = Pytest

## ***Тест 2.***

### ***Тема 5. Фреймворки Python для тестирования***

1. Прочитайте текст и установите соответствие. К каждой позиции, данной в левом столбце, подберите соответствующую позицию в правом столбце.

При разработке программного обеспечения применяют разные подходы. Чаще всего говорят о подходах на основе поведения (Behavior-driven development, BDD) и разработке через тестирование (англ. Test-driven development, TDD). Установите соответствие между названием подхода к разработке и основной идеей по его применению.

=Разработка через тестирование TDD -> техника разработки программного обеспечения, которая основывается на повторении очень коротких циклов

разработки: тесты пишут перед написанием самого кода, затем пишется код, и сразу проверяют пройдет ли он тест

= Разработка на основе поведения BDD -> поведение процесс реализации задачи начинается не с написания кода, а с описания тестов к задаче, которую предстоит реализовать на основе user story.

2. Прочитайте текст и установите соответствие. К каждой позиции, данной в левом столбце, подберите соответствующую позицию в правом столбце.

При разработке программного обеспечения на основе поведения (BDD) используют пользовательские истории (user story). Пользовательские истории создают по шаблону. Установите последовательность фраз этого шаблона.

- = Первая фраза -> Как [персона],
- = Вторая фраза -> я [хочу]
- = Третья фраза -> [чтобы что]

3. Прочитайте текст и выберите один правильный ответ

Для написания тестов используется специальный формат Gherkin. Определите, какие ключевые слова используют для описания сценария

- a. Arrange-Act-Assert
- b. =Given-When-Then
- c. Model-View-Controller

4. Прочитайте текст и установите соответствие.

К каждой позиции, данной в левом столбце, подберите соответствующую позицию в правом столбце.

Зафиксируйте названия компонент проекта Selenium и их назначение

= Pytest -> среда тестирования с открытым исходным кодом, которая поддерживает модульное тестирование, функциональное тестирование и тестиро-

вание API.

= Unittest -> стандартная среда тестирования Python, которая поставляется с пакетом Python по умолчанию

= Behave -> фреймворк для реализации бизнес-ориентированных тестов, написанных на естественном языке

5. Прочитайте текст и установите соответствие.

К каждой позиции, данной в левом столбце, подберите соответствующую позицию в правом столбце.

Для проверки правильности в Unittest используют множество методов assert. Соедините название метода и его назначение

= assertEquals() -> для проверки ожидаемого результата

= assertTrue() -> для проверки условия

= assertRaises() -> для проверки, что метод порождает исключение

6. Прочитайте текст и выберите несколько правильных ответов

Для написания тестов используется специальный формат Gherkin. Определите, какие ключевые слова используются для описания сценария

а. Assert

б. =Given

в. =When

г. =Then

7. Прочитайте текст и выберите один правильный ответ

Определите особенность фреймворка Robot Framework

а. Поддержка модульного тестирования

б. Интеграция с JUnit

в. = Основан на подходе к тестированию, основанному на ключевых словах

г. Не требует опыта программирования

8. Прочитайте текст и выберите один правильный ответ

Выберите фреймворк, который входит в число стандартных пакетов Python?

а. =Unittest

б. PyTest

в. Behave

### ***Тест 3.***

#### ***Тема 6. Тестовый фреймворк PyTest.***

1. Прочитайте текст и выберите несколько правильных ответов

Перечислите преимущества фреймворка PyTest?

а. = Простой синтаксис, лаконичность кода

б. Отсутствие необходимости в дополнительной настройке

в. Встроен в стандартную библиотеку Python

г. =Универсальный оператор assert, подробные отчеты об ошибках

2. Прочитайте текст и выберите один правильный ответ

Фикстуры в PyTest создают и запускают, когда надо передать одни и те же входные данные нескольким тестам. Выберите, каким образом через декоратор объявляют функцию фиктурой

а. =@pytest.fixture()

б. #pytest.fixture()

в. def.fixture()

3. Прочитайте текст и установите соответствие. К каждой позиции, дан-

ной в левом столбце, подберите соответствующую позицию в правом столбце.

При разработке автоматизированных тестов с помощью Pytest используют фикстуры и маркеры. Соотнесите названия с их описанием

=Фикстуры -> функции, которые создают окружение вокруг тестов

= Маркеры -> отмечают тестовые функции или блоки кода для применения определенных настроек во время выполнения тестов

4. Прочитайте текст и установите соответствие. К каждой позиции, данной в левом столбце, подберите соответствующую позицию в правом столбце.

При разработке автоматизированных тестов с помощью Pytest используют встроенные маркеры. Соотнесите названия маркеров с их назначением

= маркер skip -> позволяет пропустить выполнение теста

= маркер xfail -> используют как знак для тестов, которые рассматриваются в виде "ожидаемо неудачных"

5. Прочитайте текст и выберите один правильный ответ

Определите метод, который использован для поиска веб-элементов по локаторам в Selenium

a. find\_element(By.HTML, value)

б. = find\_element(By.ID, value)

в. find\_element(By.URL, value)

г. find\_element(By.XML, value)

6. Прочитайте текст и выберите один правильный ответ

Определите метод, с помощью которого реализуют неявное ожидание в Selenium

a. wait\_for\_element()

б. = implicitly\_wait()

- b. time.sleep()

#### ***Тест 4.***

#### ***Тема 7. Паттерн Page Object***

1. Прочитайте текст и выберите один правильный ответ

Определите, что представляет собой Page Object Model (POM) в контексте функционального тестирования веб-приложений

- a. Модель с линейными сценарии тестирования, включающие последовательность действий пользователя.
- b. = Модель, представляющая страницу в виде объектов, с которыми взаимодействует тест.
- c. Модель со сложной логикой сценария действий пользователя на веб-странице.

2. Прочитайте текст и выберите один правильный ответ

Определите, какую проблему решает применение паттерна Page Object Model при тестировании веб-приложений

- a. Уменьшение сложности пользовательских сценариев
- b. =Автоматизированное обновления локаторов элементов при изменении страницы
- c. Использование линейных сценариев в функциональном тестировании

3. Прочитайте текст и выберите один правильный ответ

Зафиксируйте, каким образом паттерн Page Object Model улучшает структуру функциональных тестов

- a. Добавление сложной логики в тестовый код.
- b. = Исключение кода теста от кода веб-страницы, разделение логики

тестирования и локаторов.

в. Использование линейных сценариев вместо модели объекта страницы.

4. Прочитайте текст и выберите один правильный ответ

Определите особенность использования паттерна Page Object Model (POM) при создании автотестов.

а. Включение логики тестов в реализацию веб-страницы.

б. =Разделение логики тестов от реализации, создание модели для каждой веб-страницы и ее элементов.

5. Прочитайте текст и выберите несколько правильных ответов

Выберите компоненты, которые включает в себя паттерн Page Object (POM)

а. =Base Page

б. =Page Object

в. =Tests

г. Base Elements

6. Прочитайте текст и выберите один правильный ответ

Определите, каким образом Page Object Model способствует поддерживаемости и устойчивости автоматизированных тестов.

а. Ограничивает использование асsertов в тестовых скриптах.

б. =Уменьшает дублирование кода путем вынесения логики работы с элементами интерфейса в отдельные классы (Page Objects).

в. Исключает необходимость использования переменных и циклов в тестовых скриптах.

Весь комплект контрольно-измерительных материалов для проверки сформированности компетенции (индикатора компетенции) размещен в закрытой части по адресу, указанному в п. 5.3

### **6.3 График текущего контроля успеваемости**

<b>Неделя</b>	<b>Темы занятий</b>	<b>Вид контроля</b>
1	Тестирование клиентской части веб-приложений на основе локаторов	
2		Практическая работа
3	Тестирование клиентской части веб-приложений на основе локаторов	
4		Практическая работа
5	Автоматизация тестирования	
6		Тест
7	Элементы автоматизации веб-приложений	
8		Практическая работа
9	Инструмент управления браузером Selenium WebDriver	
10		Практическая работа
11	Фреймворки Python для тестирования	Тест
12	Тестовый фреймворк PyTest	
13		Практическая работа
14	Тестовый фреймворк PyTest	Тест
15	Паттерн Page Object	
16		Практическая работа
17	Паттерн Page Object	Тест

### **6.4 Методика текущего контроля**

Для допуска к дифференцированному зачету студенту следует набрать не менее 55 рейтинговых баллов, которые могут быть набраны при прохождении контрольных точек курса, к которым относятся тесты и практические работы. Также необходимо посмотреть не менее 55% видео-лекций онлайн-курса. Суммарно студент может получить за семестр 100 рейтинговых баллов. При выставлении итоговой оценки по четырехбалльной системе оценивания проводят пересчет рейтинговых баллов в соответствии с п 6.1.

Формами текущего контроля определены тесты и практические работы, представленные в онлайн-курсе.

#### **Оценивание тестов**

Предусмотрено 4 теста. Каждый тест состоит из 10 вопросов, время на прохождения теста 7-10 минут.

За пройденные тесты будет начислено от 1 до 7 баллов. В результате успешного прохождения можно получить следующие баллы: за тесты № 1 и 2 по 7 баллов (суммарно 14 баллов), за тесты № 3 и 4 по 4 балла (суммарно 8 баллов).

Максимально по тестовой части можно набрать 22 баллов.

### **Оценивание практических работ**

В зависимости от уровня и полноты раскрытия темы студенты могут получить от 1 до 8 баллов за выполнение каждого практического задания. Максимально 48 баллов.

Для каждой работы установлены критерии оценивания, представленные в соответствующих разделах онлайн-курса (если, конечно, я правильно вас тут поняла).

### **Итоговый тест**

Курс завершает итоговый тест, при успешном прохождении которого студент может набрать 30 баллов. Минимальный проходной балл по итоговому тесту – 5 баллов.

### **Оценивание самостоятельной работы студентов**

При выполнении самостоятельной работы студентам требуется посмотреть видео-лекции, изучить текстовые материалы, сопровождающие каждую лекцию, внимательно прочитать методические указания по выполнению практических работ, ознакомиться с мастер-классами. Текущий контроль самостоятельной работы осуществляется посредством контроля за прохождением всех контрольных точек онлайн-курса в установленное расписание время. Критериями контроля со стороны преподавателя является полнота и своевременность выполнения практических работ, а также успешное прохождения тестовых мероприятий.

## **7 Описание информационных технологий и материально-технической базы**

<b>Тип занятий</b>	<b>Тип помещения</b>	<b>Требования к помещению</b>	<b>Требования к программному обеспечению</b>
Лекция	Лекционная аудитория	Количество посадочных мест – в соответствии с контингентом, рабочее место преподавателя с компьютером, проектор, экран, маркерная доска.	1) Windows XP и выше; 2) Microsoft Office 2007 и выше; 3) Yandex Browser или Google Chrome
Практические занятия	Аудитория	Количество посадочных мест – в соответствии с контингентом, рабочее место преподавателя с компьютером, проектор, экран, маркерная доска.	1) Windows XP и выше; 2) Microsoft Office 2007 и выше; 3) Yandex Browser или Google Chrome; 4) IDE PyCharm Community Edition
Самостоятельная работа	Помещение для самостоятельной работы	Оснащено компьютерной техникой с возможностью подключения к сети «Интернет» и обеспечением доступа в электронную информационно-образовательную среду университета.	1) Windows XP и выше; 2) Microsoft Office 2007 и выше

## **8 Адаптация рабочей программы для лиц с ОВЗ**

Адаптированная программа разрабатывается при наличии заявления со стороны обучающегося (родителей, законных представителей) и медицинских показаний (рекомендациями психолого-медико-педагогической комиссии). Для инвалидов адаптированная образовательная программа разрабатывается в соответствии с индивидуальной программой реабилитации.

## **ЛИСТ РЕГИСТРАЦИИ ИЗМЕНЕНИЙ**

<b>№ п/п</b>	<b>Дата</b>	<b>Изменение</b>	<b>Дата и номер протокола заседания УМК</b>	<b>Автор</b>	<b>Начальник ОМОЛА</b>