



**УТВЕРЖДАЮ**

Генеральный директор

ООО «Мерион Нетворкс»

Тундайкина Т.Н.

*Тундайкина Т.Н.*  
«12» мая 2026 г.

**Дополнительная профессиональная программа  
повышения квалификации**

**«Основы тестирования программного обеспечения»**

г. Москва

2026

# 1. ПОЯСНИТЕЛЬНАЯ ЗАПИСКА

## 1.1. Общие сведения о программе

Дополнительная профессиональная программа повышения квалификации «Основы тестирования программного обеспечения» (далее — Программа) разработана в соответствии с требованиями Федерального закона от 29 декабря 2012 г. № 273-ФЗ «Об образовании в Российской Федерации» и приказа Министерства образования и науки Российской Федерации от 01 июля 2013 г. № 499 «Об утверждении Порядка организации и осуществления образовательной деятельности по дополнительным профессиональным программам».

Программа ориентирована на IT-специалистов, разработчиков и аналитиков, стремящихся освоить профессиональные компетенции в области обеспечения качества программного обеспечения (QA): от принципов тестирования и составления тест-документации до тестирования API, работы с базами данных и тестирования мобильных приложений.

<b>Наименование программы</b>	Основы тестирования программного обеспечения
<b>Категория слушателей</b>	Специалисты в области IT, имеющие высшее или среднее профессиональное образование
<b>Форма обучения</b>	Заочная с применением дистанционных образовательных технологий (ДОТ)
<b>Объём программы</b>	40 академических часов
<b>Срок освоения программы</b>	Не менее 1 месяца (доступ к платформе — 2 года)
<b>Итоговый документ</b>	Удостоверение о повышении квалификации (для слушателей тарифа «Наставник»)
<b>Разработчик</b>	ООО «Мерион Нетворкс» (Merion Academy)

## 1.2. Цель программы

Цель Программы — формирование профессиональных компетенций в области тестирования программного обеспечения: освоение методологии QA, практических техник тест-дизайна, работы с тест-документацией, тестирования API и мобильных приложений, а также базовых навыков работы с SQL.

## 1.3. Задачи программы

Для достижения цели Программы решаются следующие задачи:

- ознакомить слушателей с принципами тестирования ПО, ролью QA-инженера в процессе разработки и жизненным циклом дефекта;

- сформировать навыки составления тест-документации: тест-кейсов, тест-сьютов, чек-листов и баг-репортов;
- освоить основные техники тест-дизайна: эквивалентное разбиение, граничные значения, таблицы решений, попарное тестирование;
- изучить виды и уровни тестирования: функциональное, нефункциональное (UX/UI, производительность, безопасность, доступность), регрессионное, smoke, тестирование чёрным и белым ящиком;
- освоить тестирование API: протоколы HTTP/HTTPS, архитектуры REST и SOAP, работу с инструментами Postman и Chrome DevTools;
- получить базовые навыки работы с реляционными базами данных и языком SQL (SELECT, JOIN) в контексте задач тестировщика;
- научить тестировать мобильные приложения: специфика платформ iOS и Android, использование эмуляторов и симуляторов.

#### **1.4. Требования к слушателям**

К освоению Программы допускаются лица, имеющие высшее или среднее профессиональное образование. Специальных технических знаний не требуется; рекомендуется базовое понимание принципов работы веб-приложений.

#### **1.5. Материально-техническое обеспечение**

Реализация Программы осуществляется с применением дистанционных образовательных технологий на платформе Merion Academy (lms.merionet.ru). Для обучения слушателю необходимы:

- персональный компьютер или ноутбук (ОС: Windows 10/11, Ubuntu 20.04+, macOS 12+) с процессором не ниже Intel Core i5 / AMD Ryzen 5 и оперативной памятью не менее 4 ГБ;
- программное обеспечение: браузер Google Chrome (для работы с DevTools), Postman (бесплатная версия), любой SQL-клиент (DBeaver или аналог);
- доступ к сети Интернет со скоростью не менее 10 Мбит/с.

## 2. УЧЕБНЫЙ ПЛАН

№	Наименование раздела	Всего ак. ч.	Лекции/ видео	Контроль
1	Введение в тестирование ПО. Роль QA в разработке	5	4	1
2	Функциональное тестирование: тест-кейсы, чек-листы, баг-репорты, техники тест-дизайна	13	10	3
3	Виды и уровни тестирования: нефункциональное, регрессионное, UX/UI, производительность	8	6	2
4	Тестирование API: HTTP/HTTPS, REST, SOAP, Postman, Chrome DevTools	7	6	1
5	Базы данных и SQL для тестировщика. Тестирование мобильных приложений	7	6	1
	<b>ИТОГО</b>	<b>40</b>	<b>32</b>	<b>8</b>

Примечание: лекции проводятся в формате видеоуроков. Практические занятия включают составление тест-документации, работу с Postman и SQL-запросами. Итоговый контроль — тестирование в LMS.

### **3. РАБОЧАЯ ПРОГРАММА**

#### **Раздел 1. Введение в тестирование ПО. Роль QA в разработке (5 ак. ч.)**

Содержание. Понятие качества программного обеспечения. Роль QA-инженера (тестировщика) в команде разработки: отличие QA, QC и тестирования. Жизненный цикл разработки ПО (SDLC) и место тестирования в нём. Методологии разработки: Waterfall, Agile, Scrum — влияние на процессы тестирования. Семь принципов тестирования (ISTQB). Жизненный цикл дефекта: обнаружение, регистрация, исправление, верификация, закрытие. Пирамида тестирования.

Промежуточный контроль: итоговое тестирование (порог — 80%, попыток — без ограничений).

#### **Раздел 2. Функциональное тестирование: тест-документация и техники тест-дизайна (13 ак. ч.)**

Содержание. Функциональное тестирование: понятие, цели, отличие от нефункционального. Тест-кейс: структура (ID, название, предусловия, шаги, ожидаемый результат, постусловия), атрибуты, правила написания. Тест-сьют: организация тест-кейсов, приоритизация. Исследовательское тестирование: сессионный подход, когда применяется. Чек-лист: назначение, структура, отличие от тест-кейса, примеры. Баг-репорт: обязательные поля (Summary, Steps to Reproduce, Expected/Actual Result, Severity, Priority, Environment), правила оформления. Серьёзность и приоритет дефекта. Техники тест-дизайна: эквивалентное разбиение, анализ граничных значений, таблицы решений, диаграммы переходов состояний, попарное тестирование (pairwise).

Практические работы: составление тест-кейсов для типовых форм; написание баг-репортов по предложенным сценариям; применение техник тест-дизайна. Промежуточный контроль: итоговое тестирование (порог — 80%).

#### **Раздел 3. Виды и уровни тестирования (8 ак. ч.)**

Содержание. Нефункциональные виды тестирования: обзор и классификация. UX/UI тестирование: принципы удобства использования, эвристики Нильсена. Тестирование локализации и интернационализации: форматы дат, чисел, валют, направление текста. Тестирование безопасности: базовые уязвимости (XSS, SQL Injection), инструменты проверки. Тестирование доступности: стандарты WCAG. Тестирование производительности: виды нагрузочного тестирования (load, stress, spike, soak), метрики. Тестирование методами чёрного ящика (Black Box) и белого ящика (White Box): отличия, применение. Статическое тестирование: ревью требований, инспекции, walkthrough.

Динамическое тестирование. Уровни тестирования: модульное (unit), интеграционное, системное, приёмочное (UAT). Регрессионное тестирование, smoke-тестирование, повторное тестирование (re-testing): назначение и отличия.

Промежуточный контроль: итоговое тестирование (порог — 80%).

#### **Раздел 4. Тестирование API: HTTP, REST, SOAP, Postman (7 ак. ч.)**

Содержание. Клиент-серверная архитектура: компоненты, принципы взаимодействия. Протокол HTTP/HTTPS: структура запроса и ответа, заголовки, кэш, куки. Методы HTTP: GET, POST, PUT, PATCH, DELETE — назначение и различия. Коды ответов сервера: классы 2xx, 3xx, 4xx, 5xx. Форматы данных JSON и XML: структура, сравнение. Chrome DevTools, вкладка Network: перехват и анализ запросов, фильтрация, просмотр заголовков и тела ответа. REST API: принципы архитектуры, ресурсы, эндпоинты, тестирование в Postman (коллекции, переменные окружения, тесты на JavaScript). SOAP: отличия от REST, формат XML, WSDL-файл, тестирование SOAP-сервисов.

Практическая работа: тестирование публичного REST API в Postman; анализ запросов в Chrome DevTools. Промежуточный контроль: итоговое тестирование (порог — 80%).

#### **Раздел 5. Базы данных и SQL. Тестирование мобильных приложений (7 ак. ч.)**

Содержание. Базы данных и СУБД: назначение, виды (реляционные и нереляционные). Реляционные БД: таблицы, строки, столбцы, первичные и внешние ключи. Язык SQL для тестировщика: оператор SELECT (выборка данных, условия WHERE, сортировка ORDER BY, агрегатные функции), JOIN (INNER, LEFT) для объединения таблиц. Применение SQL в тестировании: проверка данных в БД после выполнения сценариев. Тестирование мобильных приложений: специфика платформ iOS и Android, типичные дефекты. Подходы к тестированию: native, hybrid, web-приложения. Чек-лист тестирования мобильного приложения: установка, совместимость, жесты, прерывания, сеть, разрешения. Эмуляторы и симуляторы: отличия, использование Android Emulator и iOS Simulator.

Практические работы: написание SQL-запросов для проверки данных; составление чек-листа для мобильного приложения. Промежуточный контроль: итоговое тестирование (порог — 80%).

#### **Итоговая аттестация (входит в общий объём программы)**

Итоговая аттестация проводится в форме тестирования на платформе Merion Academy (lms.merionet.ru). Тест охватывает все разделы программы. Порог зачёта — 80% правильных ответов. Количество попыток — неограниченно. Оценивание: «зачёт» (80% и выше) / «незачёт» (до 79% включительно).

## 4. ОЦЕНОЧНЫЕ МАТЕРИАЛЫ. ПОРЯДОК ПРОВЕДЕНИЯ АТТЕСТАЦИИ

Оценивание освоения Программы осуществляется в форме тестирования на образовательной платформе Merion Academy (lms.merionet.ru). Применяется система «зачёт / незачёт».

### 4.1. Промежуточный контроль

По завершении каждого из 5 разделов слушатель проходит тематический тест. Параметры:

Параметр	Значение
Форма контроля	Тестирование в LMS (автоматизированная проверка)
Количество попыток	Неограниченно
Порог зачёта	≥ 80% правильных ответов
Результат «незачёт»	До 79% включительно
Система оценивания	Зачёт / Незачёт

### 4.2. Итоговая аттестация

К итоговой аттестации допускаются слушатели, успешно сдавшие все промежуточные тесты.

Параметр	Значение
Форма итоговой аттестации	Итоговое тестирование
Количество попыток	Неограниченно
Порог зачёта	≥ 80% правильных ответов
Результат «незачёт»	До 79% включительно — незачёт, передача без ограничений
Документ по итогам аттестации	Удостоверение о повышении квалификации — выдаётся слушателям тарифа «Наставник» при результате «зачёт»

## **5. УЧЕБНО-МЕТОДИЧЕСКОЕ ОБЕСПЕЧЕНИЕ**

### **5.1. Основная литература**

18. Куликов С.С. Тестирование программного обеспечения. Базовый курс. — 3-е изд. — Минск: Четыре четверти, 2023. — 312 с.
19. Савин Р. Тестирование Dot Com, или Пособие по жестокому обращению с багами в интернет-стартапах. — М.: Дело, 2010. — 312 с.
20. Myers G., Sandler C., Badgett T. The Art of Software Testing. — 3rd ed. — Wiley, 2011. — 240 p.
21. Фаулер М. Рефакторинг. Улучшение проекта существующего кода. — 2-е изд. — М.: Вильямс, 2019. — 448 с.
22. ISTQB. Certified Tester Foundation Level Syllabus v4.0. — International Software Testing Qualifications Board, 2023. — URL: <https://istqb.org>

### **5.2. Нормативно-правовые акты и стандарты**

23. Федеральный закон от 27.07.2006 № 149-ФЗ «Об информации, информационных технологиях и о защите информации» (с изм.).
24. ГОСТ Р 56920-2016. Системная и программная инженерия. Тестирование программного обеспечения. — М.: Стандартинформ, 2016.

### **5.3. Методические материалы**

25. Merion Academy. Методические материалы курса «Основы тестирования программного обеспечения». — М.: Merion Networks LLC, 2024. — <https://merionet.ru>

### **5.4. Интернет-ресурсы**

26. ISTQB. Международные стандарты тестирования — <https://istqb.org>
27. Software Testing Help — <https://www.softwaretestinghelp.com>
28. Postman Learning Center — <https://learning.postman.com>
29. Merion Academy — <https://merionet.ru>
30. LMS-платформа курса — <https://lms.merionet.ru>