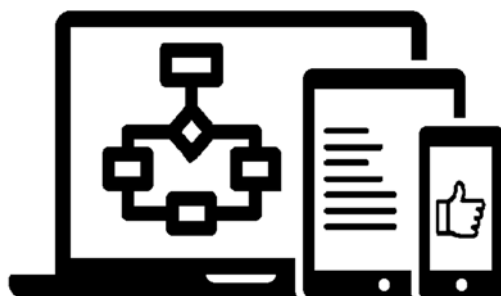


МЕЖГОСУДАРСТВЕННОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ  
ВЫСШЕГО ОБРАЗОВАНИЯ  
«БЕЛОРУССКО-РОССИЙСКИЙ УНИВЕРСИТЕТ»

Кафедра «Программное обеспечение информационных  
технологий»

# ТЕСТИРОВАНИЕ И ОТЛАДКА ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ

*Методические рекомендации к лабораторным работам  
для студентов направления подготовки  
09.03.04 «Программная инженерия»  
очной формы обучения*



Могилев 2023

УДК 621.01  
ББК 36.4  
Т87

Рекомендовано к изданию  
учебно-методическим отделом  
Белорусско-Российского университета

Одобрено кафедрой «Программное обеспечение информационных технологий» «15» декабря 2022 г., протокол № 5

Составитель ст. преподаватель Е. А. Зайченко

Рецензент канд. техн. наук, доц. И. В. Лесковец

Методические рекомендации к лабораторным работам предназначены для студентов направления подготовки 09.03.04 «Программная инженерия» очной формы обучения.

Учебно-методическое издание

## ТЕСТИРОВАНИЕ И ОТЛАДКА ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ

Ответственный за выпуск	В. В. Кутузов
Корректор	А. А. Подошевка
Компьютерная верстка	М. М. Дударева

Подписано в печать 13.01.2023 . Формат 60×84/16. Бумага офсетная. Гарнитура Таймс.  
Печать трафаретная. Усл. печ. л. 0,70. Уч.-изд. л. 0,75 . Тираж 21 экз. Заказ № 34.

Издатель и полиграфическое исполнение:  
Межгосударственное образовательное учреждение высшего образования  
«Белорусско-Российский университет».  
Свидетельство о государственной регистрации издателя,  
изготовителя, распространителя печатных изданий  
№ 1/156 от 07.03.2019.  
Пр-т Мира, 43, 212022, г. Могилев.

© Белорусско-Российский  
университет, 2023

## Содержание

Введение.....	4
1 Лабораторная работа № 1. Составление плана тестирования .....	5
2 Лабораторная работа № 2. Проектирование тест-кейсов .....	7
3 Лабораторная работа № 3. Составление документации для тестирования .....	9
4 Лабораторная работа № 4. Автоматизация тестирования .....	11
Список литературы.....	12

## Введение

Основной целью преподавания дисциплины «Тестирование и отладка программного обеспечения» является ознакомление студентов с основными видами и методами тестирования программного обеспечения при структурном и объектно-ориентированном подходе в программировании. Рассматриваемый в методических рекомендациях материал нацелен на изучение основ технологии тестирования программ и получение практических навыков разработки как ручных тестов, так и автоматизированного тестирования, а также составления документации по тестированию.

При изучении дисциплины «Тестирование и отладка программного обеспечения» студенты выполняют лабораторные работы. Перед тем как приступить к выполнению работы, необходимо изучить теоретические материалы и примеры из книги С. С. Куликова «Тестирование программного обеспечения. Базовый курс» [1]. Книга имеется в открытом доступе сети Интернет.

К защите работы студент подготавливает отчет, включающий в себя титульный лист, формулировку задания, описание исходных и результирующих данных, результаты тестирования.

Защита лабораторной работы состоит из двух частей: практической и теоретической. В практической части студент дает описание выполненной им работы, в теоретической – отвечает на вопросы по теме лабораторной работы.

При подготовке к защите студенту рекомендуется ответить на контрольные вопросы.

# 1 Лабораторная работа № 1. Составление плана тестирования

**Цель работы:** получение навыков в разработке плана тестирования.

## *Общие положения*

Тестирование программного обеспечения – процесс анализа программного средства и сопутствующей документации с целью выявления дефектов и повышения качества продукта [1].

Качество (Quality) – степень, с которой компонент, система или процесс соответствует зафиксированным требованиям и/или ожиданиям и нуждам пользователя или заказчика. В более широком смысле тестирование – это одна из техник контроля качества, включающая в себя активности по планированию работ (Test Management), проектированию тестов (Test Design), выполнению тестирования (Test Execution) и анализу полученных результатов (Test Analysis).

Верификация (Verification) – это процесс оценки системы или её компонентов с целью определения, удовлетворяют ли результаты текущего этапа разработки условиям, сформированным в начале этого этапа.

Валидация (Validation) – это определение соответствия разрабатываемого ПО ожиданиям и потребностям пользователя, требованиям к системе.

План Тестирования (Test Plan) – это документ, описывающий весь объем работ по тестированию, начиная с описания объекта, стратегии, расписания, критериев начала и окончания тестирования до необходимого в процессе работы оборудования, специальных знаний, а также оценки рисков с вариантами их разрешения.

Тест-дизайн (Test Design) – это этап процесса тестирования ПО, на котором проектируются и создаются тестовые случаи (тест-кейсы) в соответствии с определёнными ранее критериями качества и целями тестирования.

Тестовый случай (Test Case) – это артефакт, описывающий совокупность шагов, конкретных условий и параметров, необходимых для проверки реализации тестируемой функции или её части.

Дефект (defect, bug, ошибка) – ключевой термин тестирования, означающий отклонение фактического результата от ожидаемого. Для обнаружения дефекта необходимо выполнить три условия: знать фактический результат, знать ожидаемый результат, зафиксировать факт разницы между фактическим и ожидаемым результатом.

Источником ожидаемого результата является спецификация – детальное описание того, как должно работать программное обеспечение (ПО).

В общем случае любой дефект представляет собой отклонение от спецификации. Важно обнаружить эти дефекты до того, как их найдут конечные пользователи.

Тестирование можно классифицировать по очень большому количеству признаков.

Функциональные тесты базируются на функциях и особенностях, а также взаимодействии с другими системами и могут быть представлены на всех уровнях тестирования: компонентном или модульном, интеграционном, системном и

приемочном. Функциональные виды тестирования рассматривают внешнее поведение системы.

Нефункциональное тестирование описывает тесты, необходимые для определения характеристик программного обеспечения, которые могут быть измерены различными величинами.

Подробное описание видов тестирования приведено в [1, с. 64–111].

### Практическое задание

1 Самостоятельно выбрать объект реального мира для последующей разработки тестовых проверок для него.

2 Разработать различные проверки в соответствии с классификацией видов тестирования для выбранного в п. 1 объекта реального мира. Результаты внести в таблицу, составленную по образцу таблицы 1.

3 Разработать композицию тестов для первой поставки программного обеспечения, состоящей из трех модулей (модуль 1, модуль 2, модуль 3). Пример выполнения задания приведен в [1, с. 151–153].

4 Разработать композицию тестов для второй поставки программного обеспечения: исправлены заведенные дефекты, доставлена новая функциональность – модуль 4.

5 Разработать композицию тестов для третьей поставки программного обеспечения: заказчик хочет убедиться, что программное обеспечение выдержит нагрузку в 500 пользователей.

Таблица 1 – Тестовые проверки для различных видов тестирования

Объект тестирования _____		
Вид тестирования	Краткое определение вида тестирования	Тестовые проверки
Функциональное тестирование		
Тестирование безопасности		
Тестирование защищенности		
Тестирование совместимости		
Тестирование пользовательского интерфейса		
Тестирование удобства использования		
Тестирование доступности		
Тестирование производительности		
Стрессовое тестирование		
Тестирование «черного ящика»		
Автоматизированное тестирование		
Модульное тестирование		
Тестирование установки		

### ***Контрольные вопросы***

- 1 Что такое тестирование?
- 2 Что такое качество программного обеспечения?
- 3 Что такое дефект?
- 4 Назовите три условия обнаружения дефекта.
- 5 Перечислите и охарактеризуйте виды тестирования, которые зависят от объекта тестирования.
- 6 Перечислите и охарактеризуйте виды функционального тестирования.
- 7 Перечислите и охарактеризуйте виды нефункционального тестирования.
- 8 Перечислите и охарактеризуйте виды тестирования, которые зависят от глубины покрытия.
- 9 Перечислите и охарактеризуйте виды тестирования, которые зависят от тестовой активности.
- 10 Перечислите и охарактеризуйте виды тестирования в зависимости от знания кода.
- 11 Перечислите и охарактеризуйте виды тестирования в зависимости от степени автоматизации.
- 12 Перечислите и охарактеризуйте виды тестирования в зависимости от изолированности компонентов.
- 13 Перечислите и охарактеризуйте виды тестирования в зависимости от подготовленности.
- 14 Перечислите и охарактеризуйте виды тестирования в зависимости от места и времени проведения.
- 15 Перечислите и охарактеризуйте этапы процесса тестирования.
- 16 Какая композиция тестов выполняется для первой поставки программного продукта?
- 17 Какая композиция тестов выполняется для последующих поставок программного продукта?

## **2 Лабораторная работа № 2. Проектирование тест-кейсов**

**Цель работы:** получение навыков в разработке тест-кейсов.

### ***Общие положения***

План тестирования (Test Plan) – это документ, описывающий весь объем работ по тестированию, начиная с описания объекта, стратегии, расписания, критериев начала и окончания тестирования до необходимого в процессе работы оборудования, специальных знаний, а также оценки рисков с вариантами их разрешения.

Тест-дизайн (Test Design) – это этап процесса тестирования ПО, на котором проектируются и создаются тестовые случаи (тест-кейсы) в соответствии с определёнными ранее критериями качества и целями тестирования.

Тестовый случай (Test Case) – это артефакт, описывающий совокупность шагов, конкретных условий и параметров, необходимых для проверки реализации тестируемой функции или её части.

Баг/Дефект репорт (Bug Report) – это документ, описывающий ситуацию или последовательность действий, приведшую к некорректной работе объекта тестирования, с указанием причин и ожидаемого результата.

Тестовое покрытие (Test Coverage) – это одна из метрик оценки качества тестирования, представляющая из себя плотность покрытия тестами требований либо исполняемого кода.

Детализация тест-кейсов (Test Case Specification) – это уровень детализации описания тестовых шагов и требуемого результата, при котором обеспечивается разумное соотношение времени прохождения к тестовому покрытию

Время прохождения тест-кейса (Test Case Pass Time) – это время от начала прохождения шагов тест кейса до получения результата теста.

Подробное описание проектирования тест-кейсов приведено в [1, с. 112–204]. Пример выполнения практического задания приведен в [1, с. 121–142].

### **Практическое задание**

1 Получить у преподавателя спецификацию с требованиями к тестируемому приложению.

2 В зависимости от сложности бизнес-логики приложения выбрать наиболее подходящий вид рабочей тестовой документации.

3 Анализировать приложение разбить на модули и подмодули.

4 Разработать рабочую тестовую документацию для всех модулей и подмодулей приложения.

5 Указать номер тестируемой сборки, название приложения, тип выполняемой тестовой активности, период времени тестирования, ФИО тестировщика, тестовое окружение (операционная система, браузер).

6 Предусмотреть проверки пользовательского интерфейса для каждого модуля.

7 Предусмотреть общие функциональные проверки для каждого модуля.

8 В рамках каждого модуля предусмотреть функциональные проверки. Степень детализации должна соответствовать выбранному на этапе 1 типу тестовой документации.

9 Для каждой проверки указать глубину тестового покрытия с учетом выбранного на этапе 1 типа тестов документации.

### **Контрольные вопросы**

1 Какие существуют разновидности рабочей тестовой документации?

2 Что такое Test Case?

3 Какова структура описания Test Case?

4 Что содержит Идентификатор в описании Test Case?

5 Что приводится в поле Приоритет описания Test Case?

6 Что приводится в поле Требование описания Test Case?

7 Что приводится в поле Модуль и подмодуль приложения описания Test Case?

8 Что приводится в поле Заглавие описания Test Case?

9 Что приводится в поле Исходные данные, приготовления описания Test Case?



- 10 Что приводится в поле Шаги описания Test Case?
- 11 Что приводится в поле Ожидаемые результаты описания Test Case?
- 12 Для чего нужны Test Case?
- 13 Какие проверки выполняют при тестировании GUI?
- 14 Какие общие функциональные проверки выполняют для всего приложения?
- 15 Перечислите базовые проверки, применяемые для поля ввода данных.
- 16 Перечислите базовые проверки, применяемые для поля загрузки файлов.
- 17 Перечислите базовые проверки, применяемые для ввода даты.
- 18 Перечислите базовые проверки, применяемые для поля со списком.
- 19 Перечислите базовые проверки, применяемые для радиокнопки.
- 20 Перечислите базовые проверки, применяемые для чек-бокса.
- 21 Перечислите базовые проверки, применяемые для меню.
- 22 Перечислите базовые проверки, применяемые для таблиц.
- 23 Перечислите базовые проверки, применяемые для ссылок.
- 24 Перечислите базовые проверки, применяемые для сообщений.

### **3 Лабораторная работа № 3. Составление документации для тестирования**

**Цель работы:** получение навыков составления документации для тестирования.

#### ***Общие положения***

Итоговый отчет о качестве проверенного функционала является неотъемлемой частью работы, которую каждый тестирующий должен выполнить по завершении тестирования.

Итоговый отчет можно разделить на части с соответствующей информацией.

- 1 Общая информация.
- 2 Сведения о том, кто и когда тестировал программный продукт.
- 3 Тестовое окружение.
- 4 Общая оценка качества приложения.
- 5 Обоснование выставленного качества.
- 6 Графическое представление результатов тестирования.
- 7 Детализированный анализ качества по модулям.
- 8 Список самых критичных дефектов.
- 9 Рекомендации.

Далее рассмотрим подробно каждую часть итогового отчета.

Общая информация включает:

- название проекта;
- номер сборки;
- модули, которые подверглись тестированию (в случае, если тестировался не весь проект);
- виды тестов по глубине покрытия (Smoke Test, Minimal Acceptance Test, Acceptance Test), тестовые активности (New Feature Test, Regression Testing, Defect Validation);

- количество обнаруженных дефектов;
- вид рабочей тестовой документации (Acceptance Sheet, Test Survey, Test Cases).

Сведения о том, кто и когда тестировал программный продукт, включают информацию о команде тестирования с указанием контактных данных и временном интервале тестирования.

Тестовое окружение содержит ссылку на проект, браузер, операционную систему и другую информацию, конкретизирующую особенности конфигурации.

Общая оценка качества приложения выставляется на основании анализа результатов работы с приложением, количества внесенных дефектов, важности дефектов. Обязательно учитывается этап разработки проекта – то, что не критично в начале работы, становится важным при выпуске программного продукта. Уровни качества: высокое (High), среднее (Medium), низкое (Low).

Обоснование выставленного качества является наиболее важной частью отчета, т. к. здесь отражается общее состояние сборки, а именно:

- качество сборки на текущий момент;
- факторы, повлиявшие на выставление именно такого качества сборки: указание функционала, который заблокирован для проверки, перечисление наиболее критичных дефектов и объяснение их важности для пользователя или бизнеса заказчика;
- анализ качества проверенного функционала: улучшилось оно или ухудшилось по сравнению с предыдущей версией.

Подробное описание составления документации для тестирования приведено в [1, с. 205–250].

### **Практическое задание**

- 1 Составить итоговый отчет по результатам тестирования web-приложения.
- 2 Указать общую информацию о тестируемом продукте (название, номер сборки, виды выполненных тестов, количество обнаруженных дефектов, вид рабочей тестовой документации).
- 3 Указать, кто и когда тестировал программный продукт.
- 4 Описать тестовое окружение (ссылку на web-приложение, браузер).
- 5 Указать общую оценку качества протестированного приложения и подробно ее обосновать.
- 6 Произвести детальный анализ качества всех модулей протестированного приложения с аргументацией выставленных уровней качества.
- 7 Привести список пяти наиболее критичных дефектов.
- 8 Сформулировать рекомендации по улучшению качества программного продукта.

### ***Контрольные вопросы***

- 1 Какова структура итогового отчёта о результатах тестирования?
- 2 Что содержится в разделе «Общая информация»?
- 3 Что содержится в разделе «Тестовое окружение»?
- 4 Как выставляется общая оценка качества приложения?
- 5 Что содержится в разделе «Детализированный анализ качества»?
- 6 Что содержится в разделе «Рекомендации»?

## **4 Лабораторная работа № 4. Автоматизация тестирования**

**Цель работы:** получение навыков использования средств автоматизированного тестирования.

### ***Общие положения***

Автоматизированное тестирование программного обеспечения (Software Automation Testing) – это процесс верификации программного обеспечения, при котором основные функции и шаги теста, такие как запуск, инициализация, выполнение, анализ и выдача результата, выполняются автоматически при помощи инструментов для автоматизированного тестирования.

1 Оперативность – автоматизированный скрипт не сверяется с инструкциями и документацией.

2 Экономия времени – автоматизация не требует вмешательства тестирующего, в это время он может переключиться на другие задачи.

3 Повторное использование – сценарий тестирования может использоваться неоднократно.

4 Отсутствие «человеческого фактора» – тестовый сценарий не допустит оплошностей в результатах и не пропустит времени тестирования.

5 Автоматическая отчетность – результаты тестирования автоматически сохраняются и рассылаются причастным специалистам.

Вместе с тем данный подход не является панацеей и не исключает ряд недостатков.

1 Затраты – хорошие инструменты автоматизированного тестирования, как и обучение автоматизированному тестированию ПО требует вложений.

2 Однообразие – написанные тесты работают всегда одинаково, что не всегда плохо, но иногда позволяет пропустить дефект, который заметил бы живой человек.

3 Затраты на поддержку и разработку – чем сложнее приложение и чем чаще оно обновляется, тем более затратна разработка и модификация автоматизированных тестов.

4 Пропуск мелких недочетов – тесты пропускают небольшие ошибки, на проверку которых не запрограммированы.

К инструментам автоматизации тестирования относят программное обеспечение, используемое для создания, наладки, выполнения и анализа результатов работы автоматизированных тестов.

Выбор инструмента зависит от объектов тестирования и требований к сценариям тестов.

Подробное описание составления документации для тестирования приведено в [1, с. 254–266].

### **Практическое задание**

1 Адаптировать разработанные в лабораторной работе № 2 тест-кейсы для проведения автоматизированного тестирования.

2 Выполнить выбор наиболее подходящего для выполнения автоматизированного тестирования данной задачи программного средства автоматизации. Обосновать выбор.

3 Выполнить автоматизированное тестирование приложения с применением методики «Record & Platback».

4 Выполнить автоматизированное тестирование приложения с применением методики «Тестирование под управлением данными (DDT)».

5 Оформить документацию по проведенному автоматизированному тестированию.

### **Контрольные вопросы**

1 Что называется автоматизированным тестированием?

2 Какие преимущества и недостатки автоматизированного тестирования?

3 Перечислите области применения автоматизации тестирования.

4 Охарактеризуйте возможности выбранных для выполнения работы средств автоматизированного тестирования.

5 Охарактеризуйте области применения методики «Record & Platback».

6 Охарактеризуйте области применения методики «Тестирование под управлением данными (DDT)».

## **Список литературы**

1 Куликов, С. С. Тестирование программного обеспечения. Базовый курс [Электронный ресурс] / С. С. Куликов. – Минск: Четыре четверти, 2022. – 296 с. – Режим доступа : [http://svyatoslav.biz/software\\_testing\\_book/](http://svyatoslav.biz/software_testing_book/). – Дата доступа: 29.12.2022.