

Министерство науки и высшего образования РФ
Федеральное государственное бюджетное образовательное учреждение
высшего образования
«Дагестанский государственный технический университет»

Учебно-методические указания
по выполнению лабораторных работ по дисциплине
«Оценка качества программного обеспечения»
для направления подготовки
09.04.04 «Программная инженерия»
направленности «Системы искусственного интеллекта»

Махачкала 2021

1 Лабораторная работа № 1. Составление плана тестирования

Цель работы: изучить элементы плана тестирования; приобрести навыки разработки плана тестирования.

1.1 Краткие теоретические сведения

Тест-план, или план тестирования (test plan), – часть проектной документации, описывающая и регламентирующая процесс тестирования.

1.1.1 Ключевые секции тест-плана.

1.1.1.1 Перечень работ.

Эта секция включает перечень функциональных областей приложений, которые будут подвергаться тестированию.

Здесь же может быть перечень компонентов или функциональности, которые не будут тестироваться по тем или иным причинам.

1.1.1.2 Критерии качества.

Одна из наиболее важных секций, содержащая перечень критериев, по которым оценивается текущее и финальное качество продукта.

1.1.1.3 Оценка рисков.

Риск (risk) – сочетание вероятности наступления события и последствий, вызванных этим событием.

В секции приводится перечень рисков, которые могут (вполне вероятно) возникнуть в процессе работы с проектом. По каждому риску даётся оценка представляемой им угрозы и приводятся варианты выхода из ситуации.

1.1.1.4 Документация.

Здесь приводится полный перечень используемой документации, а также указывается, кто и когда должен её готовить, кому передавать и т.п.

1.1.1.5 Стратегия тестирования.

Стратегия тестирования подразумевает описание процесса тестирования с точки зрения применяемых методов, подходов, инструментальных средств и т.п.

1.1.1.6 Ресурсы.

В данной секции перечисляются необходимые для успешного проведения тестирования ресурсы:

- программные;
- аппаратные;
- человеческие;
- временные;
- финансовые.

1.1.1.7 Метрики.

Секция, в которой приводятся числовые характеристики показателей качества, способы их оценки, формулы и т.п.

1.1.1.8 Расписание и ключевые точки.

Самая простая для понимания секция. Фактически это календарь, в котором указано, что и к какому моменту должно быть сделано.

Задание

1 Приведите по пять примеров тестов для графического редактора Photoshop, которые относились бы:

- а) к уровню Smoke Test;
- б) к уровню Critical Path Test;
- в) к уровню Extended Test.

2 Приведите пример риска, который может быть отмечен при планировании тестовых испытаний. Дайте рекомендации по недопущению такой ситуации и выходу из неё в случае возникновения.

3 Перечислите основные секции тестового плана и дайте краткое пояснение того, что размещается в каждой из них.

Баллы суммируются по всем пунктам (максимальный балл равен 5) в соответствии с таблицей 1.1.

Таблица 1.1 – Критерий оценки выполнения заданий

Что оценивается	Количество правильных ответов	Количество баллов
1а. Приведено правильных примеров	2 и менее	0
	3 и более	1
1б. Приведено правильных примеров	2 и менее	0
	3 и более	1
1в. Приведено правильных примеров	2 и менее	0
	3 и более	1
2 Пример риска	Не приведён	0
	Приведён, но без рекомендаций	0,5
	Приведён с рекомендациями	1
3 Перечислено секций тестового плана	3 и менее	0
	4 и более без пояснений	0,5
	4 и более с пояснениями	1

Контрольные вопросы

- 1 Что подлежит тестированию?
- 2 Что такое тест-план?
- 3 Как определить критерии качества?
- 4 Что такое «риски» и как они оцениваются?

2 Лабораторная работа № 2. Проектирование тест-кейсов

Цель работы: приобрести практические навыки создания тестов и тест-кейсов; научиться создавать тест-кейсы для приложений.

2.1 Краткие теоретические сведения

2.1.1 Рекомендации по разработке тестов.

Начинайте с простых очевидных тестов.

Затем переходите к более сложным тестам.

Помните о граничных условиях.

Если остаётся время, занимайтесь исследовательским тестированием.

Последовательность разработки и выполнения тестов:

- простые позитивные;
- простые негативные;
- сложные позитивные;
- сложные негативные.

2.1.2 Оформление тест-кейсов.

Общие идеи по разработке тест-кейсов приведены на рисунке 2.1.

Приоритет	Связанное с тестом требование		Заглавие (суть) теста	Ожидаемый результат по каждому шагу	
UG_U 1.12	A	R97	Галерея Загрузка файла	Галерея, загрузка файла, имя со спецсимволами Приготовление: создать непустой файл с именем #\$\$^&_j_1 1. Нажать кнопку «Исходные данные, необходимые для выполнения теста» 2. Нажать кнопку «Загрузка файла» 3. Выбрать из списка подготовленный файл 4. Нажать кнопку «Выполнить» 5. Нажать кнопку «В галерею»	1. Появляется кнопка загрузки картинки 2. Появляется диалоговое окно браузера выбора файла для загрузки 3. Имя выбранного файла отображается в поле «Файл» 4. Диалоговое окно файла закрывается, в поле «Файл» отображается полное имя файла 5. Выбранный файл появляется в списке файлов галереи
Идентификатор	Модуль и подмодуль		Шаги		

Рисунок 2.1 – Элементы тест-кейсов

Задание

На основе представленного ниже набора требований сформируйте для разрабатываемых приложений:

- смюк-тест;
- чек-лист для теста критического пути;
- тест критического пути (насколько хватит времени – расписывайте идеи из чек-листа в полноценные тесты).

Требования к разрабатываемому приложению 1

Приложение должно выполнять математические вычисления.

- 1 Приложение должно работать под всеми версиями ОС Windows.
- 2 Приложение должно быть максимально похоже на стандартный калькулятор Windows (рисунок 2.2) за исключением некоторых особенностей.
- 3 Несколько приложений должны иметь возможность работать одновременно.
- 4 При запуске приложения должно отображаться окно со стандартными для калькулятора кнопками и полем ввода и отображения данных.
- 5 Для начала вычислений пользователь должен нажать кнопку «Начать».
- 6 Приложение должно позволять легко сохранять вычисления в выбранном пользователем формате.
- 7 Опционально предусматривается поддержка нескольких языков.
- 8 Приложение должно позволять выполнять вычисления сразу же после запуска.
- 9 Скорость вычислений должна быть максимально высокой.
- 10 Приложение должно позволять выполнять следующие операции: сложение, умножение, вычитание и деление чисел.
- 11 Приложение должно позволять строить графики простых функций.
- 12 Приложение должно запрашивать подтверждение («Результат не сохранён. Выйти?») в случае, если пользователь не сохранил результаты работы.

Требования к разрабатываемому приложению 2

- 1 Приложение должно работать под версиями ОС Windows 7 и Windows 8.
- 2 Несколько приложений должны иметь возможность работать одновременно, т. е. можно открыть несколько калькуляторов и вести в них независимые вычисления.
- 3 При запуске приложения должно отображаться окно с кнопками калькулятора (рисунок 2.2) и полем отображения данных.
- 4 Данные в приложение могут вводиться как с помощью кнопок приложения, так и с помощью клавиатуры.
- 5 Приложение должно позволять сохранять вычисления во внешний файл с расширением, задаваемым пользователем.
- 6 Должна быть предусмотрена поддержка английского и русского языков. Отображается тот язык, который выбран в ОС по умолчанию.
- 7 Вычисления должны производиться со скоростью не более 1 с.

8 Приложение должно позволять выполнять следующие операции: сложение, умножение, вычитание и деление чисел, взятие квадратного корня, возведение в степень, вычисление процентов, ввод отрицательного числа.



Рисунок 2.2 – Стандартный калькулятор Windows

Контрольные вопросы

- 1 В какой последовательности рекомендуется разрабатывать тесты?
- 2 Что такое смоук-тест? Приведите пример.
- 3 Что такое чек-лист?
- 4 Перечислите элементы тест-кейса.
- 5 Обязательно ли описывать ожидаемый результат в тест-кейсе и в чек-листе?

3 Лабораторная работа № 3. Составление документации для тестирования

Цель работы: приобрести практические навыки составления документации, используемой при тестировании приложений на примере отчета о тестировании.

3.1 Краткие теоретические сведения

Отчёт о результатах тестирования (test result report, TRR) – часть тестовой документации, включающая в себя описание процесса тестирования, суммарную информацию о протестированных за подотчётный период билдах, информацию о деятельности тестировщиков, а также некоторые статистические данные.

Цель написания TRR – предоставление лицам, заинтересованным в проекте, полной и объективной информации о текущем состоянии качества проекта. Эта информация выражается в конкретных фактах и цифрах.

Обычно TRR предоставляется для ознакомления всей проектной команде и заказчику.

Тестировщики не заинтересованы в приукрашивании отчётов и часто обладают более полной информацией о текущем состоянии качества продукта, чем какая бы то ни было другая часть проектной команды.

3.1.1 Структура отчёта о результатах тестирования.

Команда тестировщиков (test team).

В этой части TRR перечисляются все задействованные в процессе тестирования сотрудники с указанием занимаемой должности и роли на проекте в подотчётный период.

Описание процесса тестирования (testing process description).

В этой части TRR даётся краткое описание того, как происходило тестирование: какие использовались методы, техники, инструментальные средства и т. п.

Краткое описание (summary).

В этой части TRR даётся краткое описание того, какие билды были протестированы, есть ли в качестве приложения прогресс или регресс, есть ли какие-либо проблемы, требующие внимания руководства.

Краткое описание – важная часть отчёта, т. к. менеджеру проекта приходится просматривать огромное количество документации, и он часто принимает решение о необходимости более детального изучения отчёта как раз на основе краткого описания.

Расписание (testing timetable).

В данном разделе отчёта приводится детализированное описание того, какая работа и на протяжении какого времени выполнялась каждым тестировщиком.

Рекомендации (recommendations).

В этой части TRR следует подчеркнуть те важные моменты, на которые следует обратить внимание руководству или лидерам проектных команд. Здесь также, возможно, будет дана рекомендация на передачу проекта заказчику («передачу в продакшн»).

Статистика по ошибкам (bugs statistics).

Здесь приводится сводная таблица, содержащая информацию об ошибках, с которыми команде тестировщиков приходилось иметь дело в подотчётный период.

Список новых ошибок (new bugs found).

Здесь приводится список ошибок, обнаруженных командой тестировщиков за подотчётный период. Список ошибок легко извлечь из баг-трекинговой системы.

Статистика по всем ошибкам (all bugs statistics).

Здесь приводится сводная таблица, содержащая информацию об ошибках, с которыми команде тестировщиков приходилось иметь дело за всё время работы с проектом. Статистика по всем ошибкам также отражается в виде графика.

Задание

- 1 Разработать отчет о тестировании по заданию преподавателя.
- 2 Выполнить тест для проверки знаний.

Тест для проверки знаний

1 Отчёт о результатах тестирования – это:

- а) разновидность отчёта об ошибке;
- б) часть тестовой документации, включающая в себя описание процесса тестирования;
- в) диаграмма с указанием распределения дефектов по их важности;
- г) отчёт, подготавливаемый лидером команды разработчиков для лидера команды тестировщиков.

2 К целям написания отчёта о результатах тестирования относятся:

- а) стимулирование команды разработчиков;
- б) демонстрация преимуществ проекта перед конкурирующими проектами;
- в) предоставление заказчику экономической информации о проекте;
- г) предоставление лицам, заинтересованным в проекте, полной и объективной информации о текущем состоянии качества проекта.

3 Периодичность выпуска отчётов о результатах тестирования:

- а) ничем не определяется;
- б) отсутствует. Отчёт готовится один раз в конце проекта;
- в) определяется набором критериев, установленных в фирме для данного вида документации;
- г) определяется законодательными актами и стандартами.

4 К основным разделам отчёта о результатах тестирования относятся:

- а) шаги по воспроизведению;
- б) идентификатор;
- в) расписание;
- г) рекомендации.

5 В разделе «Описание процесса тестирования» отчёта о результатах тестирования приводится:

- а) список группы тестировщиков;
- б) список найденных дефектов;
- в) краткое описание того, как происходило тестирование: какие использовались методы, техники, инструментальные средства и т. п.;
- г) подробное описание процесса автоматизации тестирования, включая перечень тест-кейсов, журналы выполнения тестов и т. п.

6 В разделе «Краткое описание» отчёта о результатах тестирования приводится:

- а) краткое описание мнения команды тестировщиков о перспективах дальнейшего сотрудничества с данным заказчиком;
- б) краткое описание того, какие билды были протестированы, есть ли в качестве приложения прогресс или регресс, есть ли какие-либо проблемы, требующие внимания руководства;
- в) краткий перечень рекомендаций по закупке нового оборудования;
- г) краткое описание процесса разработки программного средства за подотчётный период.

7 Отчёт о результатах тестирования необходим:

- а) менеджеру проекта;
- б) лидеру команды разработчиков;
- в) системному администратору филиала;
- г) заказчику.

8 Финальный отчёт о результатах тестирования :

- а) такого отчёта нет;
- б) отчёт о результатах тестирования, создаваемый в конце работы с проектом;
- в) ещё одно название обычного отчёта о результатах тестирования;
- г) список найденных за весь период тестирования ошибок.

Контрольные вопросы

- 1 Что такое документ «Отчет о тестировании»?
- 2 Кто составляет отчет о тестировании?
- 3 Назовите основные разделы отчета о тестировании.
- 4 Кто использует отчет о тестировании?

4 Лабораторная работа № 4. Автоматизация тестирования

Цель работы: приобрести практические навыки проведения автоматизированного тестирования и использования программ для автоматизированного тестирования на примере программы Selenium IDE.

4.1 Краткие теоретические сведения

Автоматизация тестирования (test automation) – набор техник, подходов и инструментальных средств, позволяющий исключить человека из выполнения некоторых задач в процессе тестирования.

Преимущества автоматизации тестирования.

1 **Повторяемость** – все написанные тесты всегда будут выполняться однообразно, т.е. исключен «человеческий фактор». Тестирующий не пропустит тест по неосторожности и ничего не напутает в результатах.

2 **Быстрое выполнение** – автоматизированному скрипту не нужно сверяться с инструкциями и документациями, это сильно экономит время выполнения.

3 **Меньшие затраты на поддержку** – когда автоматические скрипты уже написаны, на их поддержку и анализ результатов требуется, как правило, меньшее время, чем на проведение того же объема тестирования вручную.

4 **Отчеты** – автоматически рассылаемые и сохраняемые отчеты о результатах тестирования.

5 **Выполнение без вмешательства** – во время выполнения тестов инженер-тестирующий может заниматься другими полезными делами или тесты могут выполняться в нерабочее время (этот метод предпочтительнее, т. к. нагрузка на локальные сети ночью снижена).

Недостатки автоматизации тестирования.

1 **Повторяемость** – все написанные тесты всегда будут выполняться однообразно. Это одновременно является и недостатком, т. к. тестирующий, выполняя тест вручную, может обратить внимание на некоторые детали и, проведя несколько дополнительных операций, найти дефект. Скрипт этого сделать не может.

2 **Затраты на поддержку** – несмотря на то что в случае автоматизированных тестов они меньше, чем затраты на ручное тестирование того же функционала, они все же есть. Чем чаще изменяется приложение, тем они выше.

3 **Большие затраты на разработку** – разработка автоматизированных тестов это сложный процесс, т. к. фактически идет разработка приложения, которое тестирует другое приложение. В сложных автоматизированных тестах также есть фреймворки, утилиты, библиотеки и прочее. Все это нужно тестировать и отлаживать, а это требует времени.

4 **Стоимость инструмента для автоматизации** – в случае, если используется лицензионное ПО, его стоимость может быть достаточно высока. Свободно распространяемые инструменты, как правило, отличаются более скромным функционалом и меньшим удобством работы.

5 Пропуск мелких ошибок – автоматический скрипт может пропускать мелкие ошибки, на проверку которых он не запрограммирован. Это могут быть неточности в позиционировании окон, ошибки в надписях, которые не проверяются, ошибки контролов и форм, с которыми не осуществляется взаимодействие во время выполнения скрипта.

Что автоматизировать при тестировании?

1 Труднодоступные места в системе (бэкенд-процессы, логирование файлов, запись в БД).

2 Часто используемая функциональность, риски от ошибок в которой достаточно высоки. Автоматизировав проверку критической функциональности, можно гарантировать быстрое нахождение ошибок, а значит и быстрое их решение.

3 Рутинные операции, такие как переборы данных (формы с большим количеством вводимых полей). Заполнение полей различными данными и их проверку после сохранения.

4 Валидационные сообщения. Заполнение полей некорректными данными и проверку на появление той или иной валидации.

5 Длинные end-to-end-сценарии.

6 Проверка данных, требующих точных математических расчетов.

7 Проверка правильности поиска данных.

Задание

Записать скрипт в Selenium IDE, сохранить в файл с расширением html и отправить почтой результат с указанием в теме письма текста «Selenium webinar».

После записи скрипта требуется:

- убедиться, что присутствуют необходимые команды для синхронизации;
- убедиться, что добавлены проверки, необходимые для тестирования того или иного функционала;
- несколько раз выполнить скрипт и убедиться в его работоспособности.

Требования к скрипту:

– должен содержать действия с одним из сайтов, доступных в общем доступе (варианты – в пункте «Список возможных сайтов»);

– количество команд в скрипте – от 5;

– скрипт может быть открыт на другой машине и запуститься без ошибок.

Если для определённых действий требуется аутентификация, можно использовать тестовые логин-пароль, которые прямо используются в скрипте.

– должно содержаться несколько проверок: и `assert`, и `verify` (в сопроводительном письме желательно указать, почему в одном случае используется `assert`, а в другом – `verify`).

Список возможных сайтов: www.tut.by; www.vkontakte.ru; www.onliner.by.

Повышенная сложность: www.facebook.com; www.gmail.com.

Контрольные вопросы

- 1 Что такое автоматизация тестирования?
- 2 В чем преимущества автоматизации тестирования?
- 3 В чем недостатки автоматизации тестирования?
- 4 Назовите известные Вам программы для автоматического тестирования и охарактеризуйте их.
- 5 Назовите области автоматизации тестирования.

Список литературы

- 1 Гагарина, Л. Г. Технология разработки программного обеспечения: учебное пособие / Л. Г. Гагарина. – Москва: Форум; Инфра-М, 2013. – 400 с.
- 2 Арлоу, Д. UML 2 и унифицированный процесс. Практический объектно-ориентированный анализ и проектирование / Д. Арлоу. – Москва: Символ-Плюс, 2015. – 624 с.
- 3 Савин, Р. Тестирование Dot Com, или Пособие по жестокому обращению с багами в интернет-стартапах / Р. Савин. – Москва: Дело, 2012. – 312 с.