

# **Лабораторная работа №9**

## **Разработка плана и проведение тестирования объектно-ориентированных программных средств**

### **Документирование результатов тестирования**

Цель: составить итоговый отчет о результатах тестирования приложения.

План занятия:

1. Изучить теоретические сведения.
2. Выполнить практическое задание по лабораторной работе.
3. Оформить отчет и ответить на контрольные вопросы.

#### *Теоретические сведения*

Итоговый отчет о качестве проверенного функционала является неотъемлемой частью работы, которую каждый тестировщик должен выполнить по завершению 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. ТОП-5 самых критичных дефектов.
9. Рекомендации.

Пример итогового отчета приведен на рисунке 1.

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).

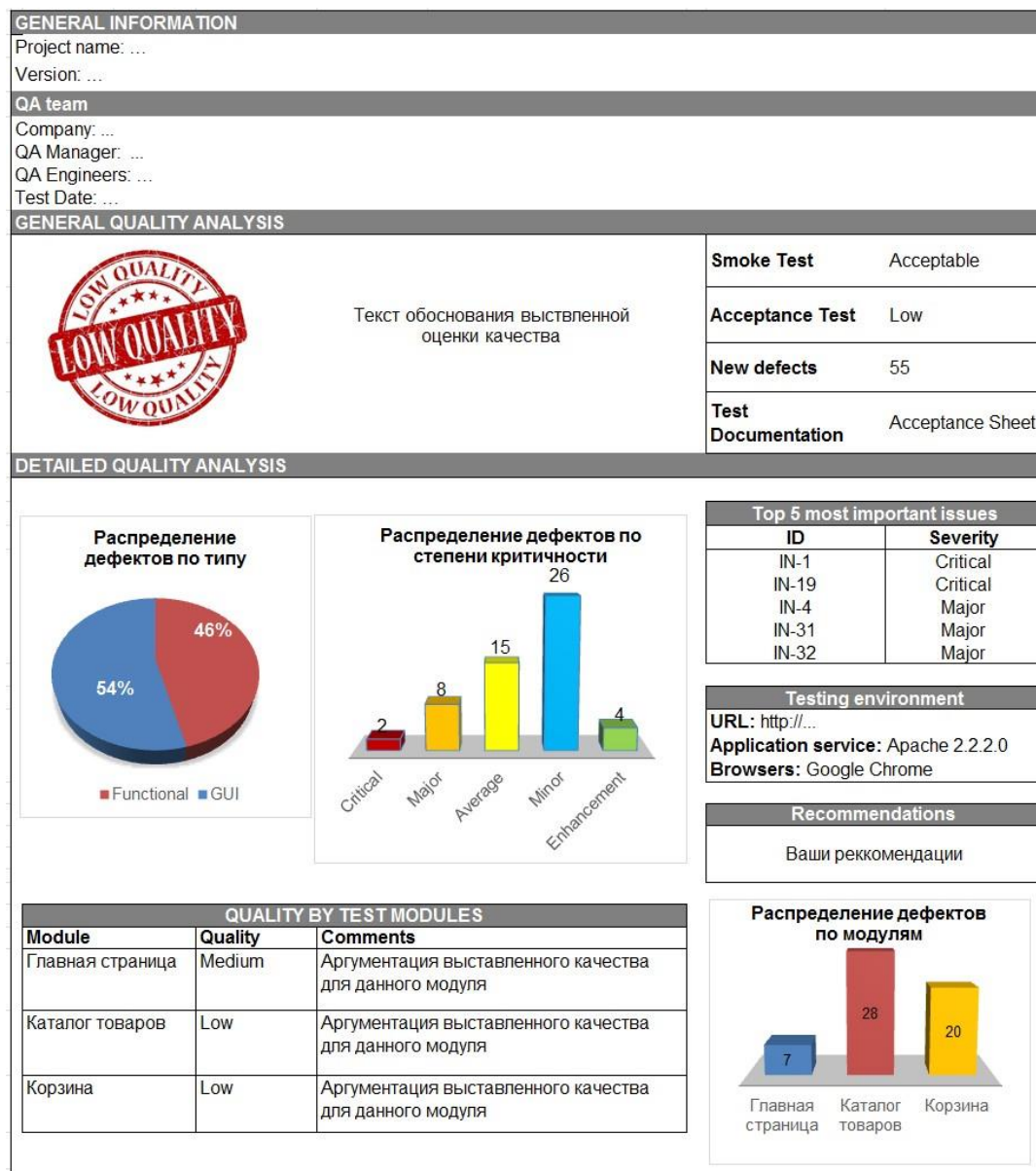


Рисунок 1 – Пример итогового отчета о результатах тестирования

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 общего впечатления от работы с приложением и внесенных дефектов (количество, важность). 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,
- если качество сборки ухудшилось, то обязательно должны быть указаны регрессионные места,
- наиболее нестабильные части функционала с указанием причин, по которым они таковыми являются.

В данном разделе показывается аналитическая работа тестировщика, наиболее слабые места и наиболее критичные дефекты, динамика изменения качества проекта.

Графическое представление результатов тестирования способствует более полному и быстрому пониманию текстовой информации (рисунок 2).

	A	B	C	D	E	F	G
1				Build	1.0		
2				Test type	NFT		
3				Test date	09-12-08-2016		
4				Tester	Ivan Ivanov		
5				Project Environment	http://...		
6				Operating System	Win 7		
7				Browser	IE 9		
8	Project Information				Test Cases Statistics		
9	Project name	Test project			Status	Quantity	Percentage (%)
10	Project URL	http://...			OK	52	2600.00%
11	Default browser	IE 9			Partially tested	0	0.00%
12					Enhancement	0	0.00%
13					Minor	0	0.00%
14					Average	0	0.00%
15					Major	0	0.00%
16					Critical	0	0.00%
17					Not available	0	0.00%
18					Not implemented	0	0.00%
19					Not tested	0	0.00%
20					Total cases	2	
21	Module	Test type		Function	Result	Issues	Comments
22	Главная страница						
23		AT	GUI		OK		
24		AT	General		OK		
25				Форма входа			
26		AT		Поле "Эл. адрес или телефон"	OK		
27		AT		Поле "Пароль"	OK		
28		AT		Кнопка "Вход"	OK		
29		AT		Чекбокс "Не выходить из системы"	OK		
30		AT		Ссылка "Забыли пароль?"	OK		
31				Форма регистрации			
32		AT		Поле "Имя"	OK		
33		AT		Поле "Фамилия"	OK		
34		AT		Поле "Эл. адрес или номер моб. телефона"	OK		
35		AT		Поле "Повторно введите ваш эл. адрес или номер моб. телефона"	OK		
36		AT		Поле "Новый пароль"	OK		
37		AT		Поле со списком "День"	OK		
38		AT		Поле со списком "Месяц"	OK		
39		AT		Поле со списком "Год"	OK		
40		AT		Ссылка "Для чего необходимо указывать дату рождения?"	OK		
41		AT		Чекбокс "Женский"/"Мужской"	OK		
42		AT		Ссылка "Условия использования"	OK		
43		AT		Ссылка "Политика использования данных"	OK		
44		AT		Ссылка "Использование файлов «cookie»"	OK		
45		AT		Кнопка "Регистрация"	OK		
46		AT		Ссылка "Создать страницу"	OK		
47				"Подвал" страницы			
48		AT		Ссылка на русскоязычную версию страницы	OK		
49		AT		Ссылка на англоязычную версию страницы	OK		
50		AT		Ссылка на арабскую версию страницы	OK		
51		AT		Ссылка на турецкую версию страницы	OK		
52		AT		Ссылка на польскую версию страницы	OK		
53		AT		Ссылка на итальянскую версию страницы	OK		
54		AT		Ссылка на вьетнамскую версию страницы	OK		
55		AT		Ссылка на немецкую версию страницы	OK		
56		AT		Ссылка на французскую версию страницы	OK		
57		AT		Ссылка на испанскую версию страницы	OK		
58		AT		Вызов всплывающего окна "Выберите ваш язык" (ссылка "...")	OK		
59		AT		Ссылка "Регистрация"	OK		

Если необходимо продемонстрировать процентное соотношение, то целесообразно использовать круговые диаграммы (например, процентное соотношение функциональных дефектов и дефектов GUI).

Столбчатые диаграммы лучше подойдут там, где важно визуализировать количество дефектов в зависимости от степени их критичности или в зависимости от локализации (распределение дефектов по модулям).

Отразить в итоговом отчёте динамику качества по всем сборкам лучше всего

удастся с помощью линейного графика.

Детализированный анализ качества по модулям.

В данной части отчета описывается более подробная информация о проверенных частях функционала, устанавливается качество каждой проверенной части функционала (модуля) в отдельности, дается аргументация выставленного уровня качества. Как правило, данный раздел отчета представляется в табличной форме. В зависимости от вида проводимых тестовых активностей, эта часть отчета будет отличаться. При оценке качества функционала на уровне Smoke теста, оно может быть либо Приемлемым (Acceptable), либо Неприемлемым (Unacceptable). Если все наиболее важные функции работают корректно, то качество всего функционала на уровне Smoke может быть оценено, как Приемлемое.

Если это релизная или предрелизная сборка, то для выставления Приемлемого качества на уровне Smoke не должно быть найдено функциональных дефектов.

В части о детализированной информации качества сборки следует более подробно описать проблемы, которые были найдены во время теста.

При оценке качества функционала на уровне Defect Validation указывается качество о проведении валидации дефектов, а именно:

- Общее количество всех дефектов, поступивших на проверку.
- Количество неисправленных дефектов и их процент от общего количества.
- Список дефектов, которые не были проверены и причины, по которым этого не было сделано.
- Наглядная таблица с неисправленными дефектами.

По вышеуказанным результатам выставляется качество теста. Если процент неисправленных дефектов  $< 10\%$ , то качество Приемлемое (Acceptable), если  $> 10\%$ , то качество Неприемлемое (Unacceptable).

При оценке качества функционала на уровне New Feature Test (полный тест нового функционала) качество отдельно проверенного функционала может быть: Высокое (High), Среднее (Medium), Низкое (Low).

Важно отдельно указывать информацию о качестве каждого модуля нового функционала с аргументацией выставленной оценки.

При оценке качества функционала на уровне Regression Testing нужно анализировать динамику изменения качества проверенной функциональности в сравнении с более ранними версиями сборки. Для этого приводится сравнительная характеристика каждой из частей функционала в сравнении с предыдущими версиями сборки, даются ясные пояснения о выставлении соответствующего качества каждой функции в отдельности. Также как и у предыдущего вида тестов, качество этих может быть: Высокое (High), Среднее (Medium), Низкое (Low).

ТОП-5 самых критичных дефектов содержит список ссылок на наиболее критичные дефекты с указанием их названия и уровня критичности.

Рекомендации включают краткую информацию о всех проблемах, характерных сборке, с пояснениями, насколько оставшиеся проблемы являются критичными для конечного пользователя. Обязательно указывают функционал и дефекты, скорейшее исправление которых является наиболее приоритетным. Кроме того, если сборка является релизной или предрелизной, то любое ухудшение качества является критичным и важно это обозначить.

*Практическое задание:*

1. Составить итоговый отчет по результатам тестирования web приложения

из лабораторной работы 8.

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. Графически (в виде круговой диаграммы) отразить процентное соотношение дефектов GUI и функциональных дефектов.

7. Графически (в виде столбчатой диаграммы) отразить распределение дефектов по различным степеням критичности.

8. Графически (в виде столбчатой диаграммы) отразить распределение дефектов по модулям.

9. Произвести детальный анализ качества всех модулей протестированного приложения с аргументацией выставленных уровней качества.

10. Привести список пяти наиболее критичных дефектов.

11. Сформулировать рекомендации по улучшению качества программного продукта.

12. Оформить отчет и защитить лабораторную работу.

Содержание отчета:

1. Цель работы.

2. Итоговый отчет о результатах тестирования web приложения.

3. Выводы по работе.

Контрольные вопросы:

1. Какая структура итогового отчета 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. Как обосновать выставленную оценку качества?

6. Для чего используется графическое представление результатов тестирования в итоговом отчете?

7. Что содержится в разделе Детализированный анализ качества?

8. Что содержится в разделе Рекомендации?