

**Уважаемые студенты!**  
**Задание на учебную практику**

1. Ознакомиться с теоретическим материалом
2. Выполнить задание. Сохранить проект в папке «ДП<sub>11</sub>-20 ФИО»
3. Оформить ОТЧЕТ в Word
4. ОТПРАВИТЬ ПРЕПОДАВАТЕЛЮ НА ПОЧТУ [v.vika2014@mail.ru](mailto:v.vika2014@mail.ru) тел

0721744922

4 Дата предоставления электронного оформления отчета (формат \*.doc)

**Лабораторная работа**  
**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.***

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 Идентификатор	A R9 Галерея Загрузка файла Модуль и подмодуль	Галерея, загрузка файла, имя со спецсимволами Приготовление: создать непустой файл с именем #\$\$^&.jpg 1. Нажать кнопку «Исходные данные, необходимые для выполнения теста» 2. Нажать кнопку «Исходные данные, необходимые для выполнения теста» 3. Выбрать из списка подготовленный файл 4. Нажать кнопку «Исходные данные, необходимые для выполнения теста» 5. Нажать кнопку «Исходные данные, необходимые для выполнения теста»	1. Появляется кнопка загрузки картинки 2. Появляется диалоговое окно браузера выбора файла для загрузки 3. Имя выбранного файла появляется в поле «Файл» 4. Диалоговое окно файла закрывается, в поле «Файл» появляется полное имя файла 5. Выбранный файл появляется в списке файлов галереи
		Шаги	

Рисунок 2.1 – Элементы тест-кейсов

### **Задание**

На основе представленного ниже набора требований сформируйте для разрабатываемых приложений:

- смоук-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?

