

## ЛАБОРАТОРНАЯ РАБОТА №4

### РАЗРАБОТКА ТЕСТ ПЛАНА

#### Планирование тестирования

*Цель работы:* изучить классификацию видов тестирования, сформировать практические навыки генерации тестов различных видов, научиться планировать тестовые активности в зависимости от специфики поставляемой на тестирование функциональности.

#### **Теоретические сведения**

Тестирование – процесс, направленный на оценку корректности, полноты и качества разработанного программного обеспечения.

Тестирование можно классифицировать по очень большому количеству признаков. Далее приведен обобщенный список видов тестирования по различным основаниям.

#### **Типы тестов по покрытию (по глубине)**

**Smoke test** – тестирование системы для определения корректной работы базовых функций программы в целом, без углубления в детали. При проведении теста определяется пригодность сборки для дальнейшего тестирования.

**Minimal Acceptance Test (MAT, Positive test):** тестирование системы или ее части только на валидных данных (валидные данные – это данные, которые необходимо использовать для корректной работы модуля/функции). При тестировании проверяется правильная работа всех функций и модулей с валидными данными.

Для крупных и сложных приложений используется ограниченный набор сценариев и функций.

**Acceptance Test (AT):** полное тестирование системы или ее части как на корректных, так и на некорректных данных/сценариях. Вид теста, направленный на подтверждение того, что приложение может использоваться по назначению при любых условиях.

Тест на этом уровне покрывает все возможные сценарии тестирования:

- проверку работоспособности модулей при вводе корректных значений;
- проверку при вводе некорректных значений;
- использование форматов данных отличных от тех, указанных в требованиях;
- проверку исключительных ситуаций, сообщений об ошибках;
- тестирование на различных комбинациях входных параметров;
- проверку всех классов эквивалентности;
- тестирование граничных значений интервалов;
- сценарии не предусмотренные спецификацией и т.д.

#### **Тестовые активности (типы тестов по покрытию (по ширине)):**

**Defect Validation** – проверка результата исправления дефектов. Включает в себя проверку на воспроизводимость дефектов, которые были исправлены в новой сборке продукта, а также проверку того, что исправление не повлияло на ранее работавшую функциональность

**New Feature Test (NFT, AT of NF)** – определение качества поставленной на тестирование новой функциональности, которая ранее не тестировалась. Данный тип тестирования включает в себя: проведение полного теста (AT) непосредственно новой функциональности; тестирование новой функциональности на соответствие документации; проверку всевозможных взаимодействий ранее реализованной функциональности с новыми модулями и функциями.

**Regression testing (регрессионное тестирование)** – проводится с целью оценки качества ранее реализованной функциональности. Включает в себя проверку стабильности ранее реализованной функциональности после внесения изменений, например добавления новой функциональности, исправление дефектов, оптимизация кода, разворачивание приложения на новом окружении. Регрессионное тестирование может быть проведено на уровне Smoke, MAT или AT.

### **Типы тестов по знанию коду**

**Черный ящик** – тестирование системы, функциональное или нефункциональное, без знания внутренней структуры и компонентов системы. У тестировщика нет доступа к внутренней структуре и коду приложения либо в процессе тестирования он не обращается к ним.

**Белый ящик** – тестирование основанное на анализе внутренней структуры компонентов или системы. У тестировщика есть доступ к внутренней структуре и коду приложения.

**Серый ящик** – комбинация методов белого и черного ящика, состоящая в том, что к части кода архитектуры у тестировщика есть, а к части кода – нет.

### **Типы тестов по степени автоматизации**

**Ручное** – тестирование, в котором тест-кейсы выполняются тестировщиком вручную без использования средств автоматизации.

**Автоматизированное** – набор техник, подходов и инструментальных средств, позволяющий исключить человека из выполнения некоторых задач в процессе тестирования. Тест-кейсы частично или полностью выполняет специальное инструментальное средство.

### **Типы тестов по изолированности компонентов**

**Unit/component (модульное)** – тестирование отдельных компонентов (модулей) программного обеспечения.

**Integration (интеграционное)** – тестируется взаимодействие между интегрированными компонентами или системами.

**System (системное)** – тестируется работоспособность системы в целом с целью проверки того, что она соответствует установленным требованиям.

### **Типы тестов по подготовленности.**

**Интуитивное тестирование** выполняется без подготовки к тестам, без определения ожидаемых результатов, проектирования тестовых сценариев.

**Исследовательское тестирование** – метод проектирования тестовых сценариев во время выполнения этих сценариев. Тестировщик совершает проверки, продумывает их, придумывает новые проверки, часто использует для этого полученную информацию.

**Тестирование по документации** – тестирование по подготовленным тестовым сценариям, руководству по осуществлению тестов.

### **Типы тестов по месту и времени проведения**

**User Acceptance Testing (UAT) (приемочное тестирование)** – формальное тестирование по отношению к потребностям, требованиям и бизнес процессам пользователя, проводимое с целью определения соответствия системы критериям приёмки и дать возможность пользователям, заказчикам или иным авторизованным лицам определить, принимать систему.

**Alpha Testing (альфа-тестирование)** – моделируемое или действительное функциональное тестирование, выполняется в организации, разрабатывающей продукт, но не проектной командой (это может быть независимая команда тестировщиков, потенциальные пользователи, заказчики). Альфа тестирование часто применяется к коробочному программному обеспечению в качестве внутреннего приемочного тестирования.

**Beta Testing (бета-тестирование)** – эксплуатационное тестирование потенциальными или существующими клиентами/заказчиками на внешней стороне (в среде, где продукт будет использоваться) никак связанными с разработчиками, с целью определения действительно ли компонент или система удовлетворяет требованиям клиента/заказчика и вписывается в бизнес-процессы. Бета-тестирование часто проводится как форма внешнего приемочного тестирования готового программного обеспечения для того, чтобы получить отзывы рынка.

### **Типы тестов по объекту тестирования**

**Functional testing (функциональное тестирование)** – это тестирование, основанное на анализе спецификации, функциональности компонента или системы. Функциональным можно назвать любой вид тестирования, который согласно требованиям проверяет правильную работу.

**Safety testing (тестирование безопасности)** – тестирование программного продукта с целью определить его безопасность (безопасность

– способность программного продукта при использовании оговоренным образом оставаться в рамках приемлемого риска причинения вреда здоровью, бизнесу, программам, собственности или окружающей среде.

**Security testing (тестирование защищенности)** – это тестирование с целью оценить защищенность программного продукта. Тестирование защищенности проверяет фактическую реакцию защитных механизмов, встроенных в систему, на проникновение.

**Compatibility testing (тестирование совместимости)** – процесс тестирования для определения возможности взаимодействия программного продукта, проверка работоспособности приложения в различных средах (браузеры и их версии, операционные системы, их типа, версии и разрядность)

#### Виды тестов:

- ✓ кросс-браузерное тестирование (различные браузеры или версии браузеров)
- ✓ кросс-платформенное тестирование (различные операционные системы или версии операционных систем)

**Нефункциональное тестирование** – это проверка характеристик программы. Иначе говоря, когда проверяется не именно правильность работы, а какие-либо свойства (внешний вид и удобство пользования, скорость работы и т.п.).

**1. Тестирование пользовательского интерфейса (GUI)** – тестирование, выполняемое путем взаимодействия с системой через графический интерфейс пользователя.

- ✓ навигация
- ✓ цвета, графика, оформление
- ✓ содержание выводимой информации
- ✓ поведение курсора и горячие клавиши
- ✓ отображение различного количества данных (нет данных, минимальное и максимальное количество)
- ✓ изменение размеров окна или разрешения экрана

**2. Тестирование удобства использования (Usability Testing)** – тестирование с целью определения степени понятности, легкости в изучении и использовании, привлекательности программного продукта для пользователя при условии использования в заданных условиях эксплуатации.

- ✓ визуальное оформление
- ✓ навигация
- ✓ логичность

**3. Тестирование доступности (Accessibility testing)** – тестирование, которое определяет степень легкости, с которой пользователи с ограниченными способностями могут использовать систему или ее компоненты.

**4. Тестирование интернационализации** – тестирование способности продукта работать в локализованных средах (способность изменять элементы интерфейса в зависимости от длины и направления текста, менять сортировки/форматы под различные локали (набор параметров, определяющий региональные настройки пользовательского интерфейса, такие как: язык, страна, часовой пояс, набор символов и т.п.) и другое).

Интернационализация – это процесс, упрощающий дальнейшую адаптацию продукта

к языковым и культурным особенностям региона, отличного от того, в котором разрабатывался продукт. Это адаптация продукта для потенциального использования практически в любом месте, Интернационализация производится на начальных этапах разработки, в то время как локализация — для каждого целевого языка.

**5. Тестирование локализации (Localization testing)** – тестирование, проводимое с целью проверить качество перевода продукта с одного языка на другой.

**6. Тестирование производительности или нагрузочное тестирование** – процесс тестирования с целью определения производительности программного продукта.

Виды тестов:

✓ нагрузочное тестирование (Performance and Load testing) – вид тестирования производительности, проводимый с целью оценки поведения компонента или системы при возрастающей нагрузке, например количестве параллельных пользователей и/или операций, а также определения какую нагрузку может выдержать компонент или система;

✓ объемное тестирование (Volume testing) – позволяет получить оценку производительности при увеличении объемов данных в базе данных приложения;

✓ тестирование стабильности и надежности (Stability / Reliability testing) – позволяет проверять работоспособность приложения при длительном (многочасовом) тестировании со средним уровнем нагрузки.

✓ стрессовое тестирование (Stress testing) – вид тестирования производительности, оценивающий систему или компонент на граничных значениях рабочих нагрузок или за их пределами, или же в состоянии ограниченных ресурсов, таких как память или доступ к серверу.

**7. Тестирование требований (Requirements testing)** – проверка требований на соответствие основным характеристикам качества.

**8. Тестирование прототипа (Prototype testing)** – метод выявления структурных, логических ошибок и ошибок проектирования на ранней стадии развития продукта до начала фактической разработки.

**9. Тестирование установки (Installability testing) и лицензирования** – процесс тестирования устанавливаемости программного продукта.

Виды тестов:

✓ формальный тест программы установки приложения (проверка пользовательского интерфейса, навигации, удобства пользования, соответствия общепринятым стандартам оформления);

✓ функциональный тест программы установки;

✓ тестирование механизма лицензирования и функций защиты от пиратства;

✓ проверка стабильности приложения после установки.

**10. Тестирование на отказ и восстановление (Failover and Recovery Testing)** – тестирование при помощи эмуляции отказов системы или реально вызываемых отказов в управляемом окружении.

**Тестирование программного продукта включает следующие этапы:**

1. Изучение и анализ предмета тестирования.
2. Планирование тестирования.
3. Исполнение тестирования.

**Изучение и анализ предмета тестирования** начинается еще до утверждения спецификации и продолжается на стадии разработки (кодирования) программного обеспечения. Конечной целью этапа изучение и анализ предмета тестирования является получение ответов на два вопроса:

- функциональности которые предстоит протестировать,
- принцип работы функциональностей.

**Планирование тестирования** происходит на стадии разработки (кодирования) программного обеспечения. На стадии планирования тестирования перед тестирующим стоит задача поиска компромисса между объемом тестирования, который возможен в

теории, и объемом тестирования, который возможен на практике. На данной стадии необходимо ответить на вопрос: как будем тестировать? Результатом планирования тестирования является тестовая документация.

**Выполнение тестирования** происходит на стадии тестирования и представляет собой практический поиск дефектов с использованием тестовой документации, составленной ранее.

**Для всех программных продуктов выполняют следующие типы тестов и их композиции.**

**Для первого билда (Билд** (от англ. *to build* - сооружать, строить) - конечный результат компиляции программы с уникальным номером версии сборки.) рекомендуется проводить Smoke+AT готовой функциональности: поверхностное тестирование (Smoke Test) выполняется для определения пригодности сборки для дальнейшего тестирования; полное тестирование системы или ее части как на корректных, так и на некорректных данных/сценариях (Acceptance Test, AT) позволяет обнаружить дефекты и внести запись о них в багтрекингтовую систему.

**Для последующих билдов композиции тестов могут быть следующими:**

- Если не была добавлена новая функциональность, то: DV+MAT. Т.е., выполняется проверка исправления дефектов программистом (Defect Validation, DV), а также проверка работоспособности остальной функциональности после исправления дефектов на позитивных сценариях (Minimal Acceptance Test, MAT).

- Если была добавлена новая функциональность, то: Smoke+DV+NFT+Regression Test. В частности, выполняется поверхностное тестирование (Smoke Test), проверка исправления дефектов программистом (Defect Validation, DV), тестирование новых функциональностей (New Feature Testing, NFT), проверка старых функциональностей, т.е. регрессионное тестирование (Regression Test).

- Если была добавлена новая функциональность, то возможен также вариант: DV+NFT+Regression test, т.е. без выполнения Smoke Test.

В зависимости от типа и специфики приложения (web, desktop, mobile) выполняют специализированные тесты (например, кроссбраузерное или кроссплатформенное тестирование, тестирование локализации и интернационализации и др.).

### *Порядок выполнения работы*

1. **Получить задание у преподавателя.**
2. Выполнить генерацию тестов различных видов для конкретного объекта реального мира (пример приведен на рисунке 1).
3. Спланировать тестовые активности для следующих задач:
  - 3.1 Поставлен на тестирование модуль 1, модуль 2, модуль 3.
  - 3.2 Проведены исправления (fix) для заведенных дефектов, доставлена новая функциональность – модуль 4.
  - 3.3 Заказчик решил расширять рынки сбыта и просит осуществить поддержку для Великобритании (кроме уже существующей Беларуси).
  - 3.4 Заказчик хочет убедиться, что ПО держит нагрузку в 2000 пользователей.
4. Оформить отчет и защитить лабораторную работу.

## Пример выполнения лабораторной работы

Необходимо составить тестовый план для объекта «Карандаш».  
Пример тестового плана для объекта карандаш представлен на рисунке 1.1.

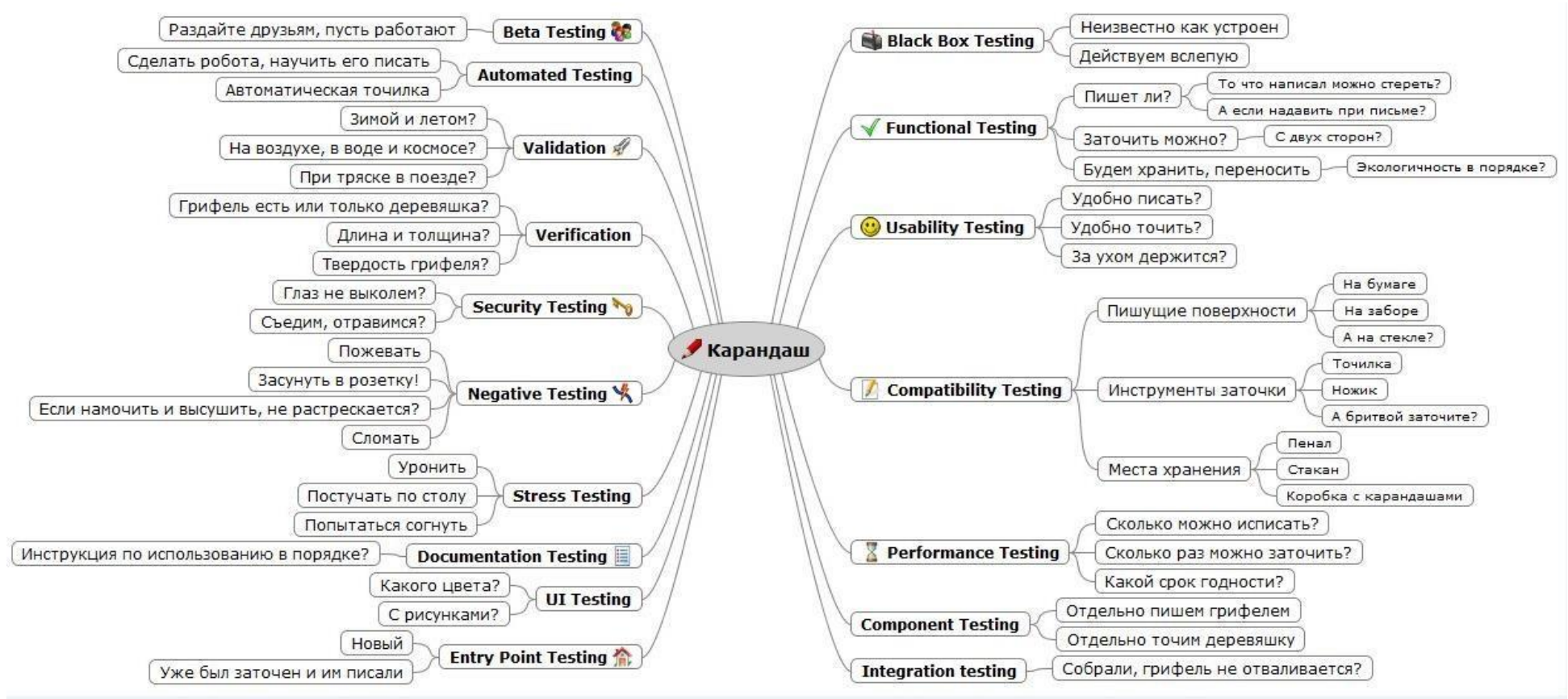


Рисунок 1.1 – Пример генерации тестов различных видов для объекта «Карандаш»

## Тест план Тестирование Блокнота AkelPad 4.8.1.0 x32

### 1.ID

#### Тестирование Блокнота AkelPad 4.8.1.0 x32

### 2. Введение (Introduction)

Цель данного документа создание тест плана приложения для PC Блокнот AkelPad x32 версии 4.8.1.0. Представленный учебный тест план оценивает подходы к тестированию данного продукта. Основное назначение данного тест плана является усвоение навыков проверки качества любого программного обеспечения.

### 3. Объекты тестирования (Test Items)

Проведение функционального тестирования:

- Проверка Однооконного (SDI), многооконного (MDI) и псевдо-многооконного режимов (PMDI);
- Полнота поддержки Unicode строк на Unicode системах (NT/2000/XP/2003/Vista/Seven);
- Работа с кодировками Unicode (UTF-8, UTF-16LE, UTF-16BE, UTF-32LE, UTF-32BE);
- Работа с любой кодовой страницей, установленной в системе;
- Предпросмотр открытия файлов;
- Блочное выделение текста;
- Многоуровневый откат действий;
- Быстрые поиск/замена строк текста;
- Запоминание кодировки и позиции каретки в файле;
- Печать и предпросмотр;
- Поддержка языковых модулей;
- Поддержка плагинов (подсветка синтаксиса, сворачивание блоков, авто-дополнение, запуск скриптов, клавиатурные макросы и много больше).

### 4. Что будет тестироваться? (Features to be tested)

Функции Блокнота, с точки зрения пользователя, что будут тестироваться:

Открытие и закрытие приложения AkelPad 4.8.1.0 x32

Создание документов

Сохранение документов

Открытие ранее созданных документов

Операции с текстом

Поиск по тексту

Шрифт и перенос по словам

Горячие клавиши

Контекстное меню

Пользовательское меню

Подсветка синтаксиса

Выполнение базовых плагинов

Работа с кодировками

Печать документа

Языковые модули

### 5. Что не будет тестироваться? (Features not to be tested)

Тестирование не будет затрагивать справочный материал, так как в данной версии программы он находится в постоянной разработке.

Тестирование сторонних плагинов так как это не обязательное дополнение к этой программе.

### 6. Подход к тестированию (Approach)

По доступу к исходному коду – Black box

По запуску кода – динамическое тестирование

По степени автоматизации – ручное

По объекту тестирования :

- функциональное
- нефункциональное

По требованиям

- позитивное
- негативное

По степени подготовленности – интуитивное тестирование (ad hoc)

В качестве тестовых данных будет использовано:

- набор текстовых данных объединенных в текстовые документы и имеющих следующий вид (\*.txt, \*.css, \*.html)

#### 7. Критерии прохождения тестов (**Item pass/fail criteria**)

Все тест кейсы, имеющие высокий приоритет, закрыты с результатом «pass/пройден».

Тестовое покрытие проверено и является достаточным, где критерий достаточности составляет не менее 99% покрытия требований тестами.

Тест репорт составлен и утвержден тест лидом и заказчиком.

#### 8. Критерии прерывания и продолжения тестирования (**Suspension criteria and resumption requirements**)

Критерий прерывания тестирования:

- появление и занесение в баг-трекиговую систему блокирующих багов.

Критерием продолжения тестирования:

- закрытие блокирующего бага в баг-трекиговой системе.

#### 9. Результаты проведения тестирования (**Test deliverables**)

Тест план, тест кейсы, тест репорт.

#### 10. Задачи для проведения тестирования (**Testing tasks**)

Задача	Расположение
Написание тест плана	Создание тест плана, обязанности
Написание тест кейсов	Объекты тестирования, обязанности
Разработка критериев успешности тестирования	Критерии успешности тестирования
Проведение тестирования и оценка результатов	Подход к тестированию, обязанности
Создание отчетов о результатах тестирования	Результаты проведения тестирования

#### 11. Технические требования (**Environmental needs**)

Тестирование приложения будет происходить на следующих операционных системах:  
Windows XP/2003/7;

#### 12. Обязанности (**Responsibilities**)

№ п/п	Роль	Функционал и Обязанности	Ответственный
1	<b>Test engineer</b>	Открытие и закрытие приложения	<b>Test engineer 1</b>
2	<b>Test engineer</b>	Создание документов	



3	<b>Test engineer</b>	Сохранение документов	
4	<b>Test engineer</b>	Открытие ранее созданных документов	
5	<b>Test engineer</b>	Операции с текстом	
6	<b>Test engineer</b>	Поиск по тексту	<b>Test engineer 2</b>
7	<b>Test engineer</b>	Шрифт и перенос по словам	
8	<b>Test engineer</b>	Горячие клавиши	
9	<b>Test engineer</b>	Контекстное меню	<b>Test engineer 3</b>
10	<b>Test engineer</b>	Пользовательское меню	
11	<b>Test engineer</b>	Подсветка синтаксиса	
12	<b>Test engineer</b>	Выполнение базовых плагинов	<b>Test engineer 4</b>
13	<b>Test engineer</b>	Работа с кодировками	<b>Test engineer 5</b>
14	<b>Test engineer</b>	Печать документа	<b>Test engineer 6</b>
15	<b>Test engineer</b>	Языковые модули	
16	<b>Test engineer</b>	Тестирование удобства использования	
17	<b>Test engineer</b>	Локализационное тестирование	<b>Test engineer 7</b>
			<b>Test engineer 8</b>

### 13.Необходимые компетенции и тренинги (**Staffing and training needs**)

Для выполнения поставленных учебных задач необходимо обладать следующими знаниями и умениями:

- знание и умение применения на практике стандарта IEEE-829;
- знание и умение применить на практике основных техник тест дизайна
- знание различных типов тестирования в том числе функционального и нефункционального.

### 14.Расписание/ срок сдачи (**Schedule**)

Срок окончания всех работ и сдачи проекта – 1/08/2018 19:00:00

### 15.Риски (Risks and contingencies)

Возможные риски во время тестирования:

- Недостаточное количество кадровых ресурсов для тестирования приложения в установленные сроки.
- Введение военного положения в стране.
- Задержки в сроках выполнения работ, вызванные другими обстоятельствами непреодолимой силы.

### 16. Утверждение (**Approvals**)

Test Lead	
<b>Team Lead</b>	
<b>Test engineer 1</b>	
<b>Test engineer 2</b>	
<b>Test engineer 3</b>	
<b>Test engineer 4</b>	
<b>Test engineer 5</b>	
<b>Test engineer 6</b>	
<b>Test engineer 7</b>	

Тест план для интернет-магазина  
<http://wall-decor.com.ua>

## Содержание

▪ Изменения документа	3
▪ Введение	4
▪ Тестовое окружение	4
▪ Тестируемый функционал сайта	5
▪ Конфигурации	5
▪ Процесс тестирования	5
▪ Критерии начала и окончания тестирования	5
▪ План работ	6
▪ Риски и ограничения	6
▪ Тестовая документация	6

## Изменения документа

Дата	Автор	Вид изменений
03.04.2015	Карпенко А. Д.	Создание
04.04.2015	Карпенко А. Д.	Создание
06.04.2015	Божко А. В.	Редактирование

## Введение

Целью составления данного Тест Плана является описание процесса тестирования интернет-магазина «<http://wall-decor.com.ua>».

**Объект тестирования:** сайт wall-decor.com.ua.

Предполагается провести:

Для отдельных полей:

- Позитивное тестирование приложения (корректные шаги, корректные данные).
- Негативное тестирование (подразумевает введение некорректных данных).

Для всей системы:

- Функциональное тестирование;
- Кросс-браузерное тестирование;
- Юзабилити тестирование;
- Тестирование пользовательского интерфейса.

Тестовое окружение:

<b>Браузеры:</b>	Mozilla Firefox, Chrome, Safari, Opera, Maxtone, IE последних версий.
<b>Операционные системы:</b>	Windows 7, Windows 8.
<b>Разрешение экрана:</b>	1366x768; 1280x800; 1280x1024; 1680x1050; 1929x1080.
<b>Отображение на дисплеях мобильных устройств с расширением:</b>	480x800; 640x960; 768x1280; 1024x768; 1366x768.

## Тестируемый функционал сайта:

Будет протестирована следующая функциональность сайта:

- каталог товаров – приоритет высокий;
- пользовательская корзина- приоритет высокий;
- регистрационная форма – приоритет высокий;
- логин;
- форма отправки заказа – приоритет высокий;
- профайл пользователя – приоритет низкий.

Не будет проведено нагрузочное и тестирование безопасности в виду отсутствия необходимых ресурсов.

Конфигурации:

<b>Персональный компьютер</b>	Windows 7 SP1
<b>Отображение сайта на мобильных устройствах</b>	HUAWEI G510-0010 Android 4.1.1 HTC Desire SV Samsung Galaxy Star Plus GT-S7262 Samsung Galaxy S3 mini Iphone 5 IPad

## Процесс тестирования:

Для максимального покрытия сайтов предполагается составление чек-листов и тест-кейсов.

Критерии начала и окончания тестирования:

Тестирование может быть начато, если выполнены следующие условия:

1. Готова и утверждена необходимая документация;
2. Тестируемый функционал окончен и готов для передачи в тестирование.

Тестирование окончено, если выполнены следующие условия:

1. все найденные дефекты задокументированы.

## План работ

Задача	Время	Дата начала	Дата окончания
Составление тест плана и чек- листа	4 час.	02.04.2015	02.04.2015
Корректировка тест плана и чек-листа	1 час.		
Выполнение тестов	IE 8-11 1 час Chrome 0,5 часа Firefox 0,5 часа Opera 0,5 часа Safari 0,5 часа	03.04.2015	03.04.2015
Написание баг-репорта	2 часа	03.04.2015	03.04.2015

### Риски и ограничения:

- ввиду ограниченности ресурсов, гарантировать корректное отображение сайта можем только на устройствах, которые указаны в таблице «Конфигурации».

Тестовая документация:

После окончания тестирования планируется наличие таких документов:

- тест-план;
- чек-лист;
- баг-репорт.

**Что надо тестировать?** - описание объекта тестирования: системы, приложения, оборудования

**Что будете тестировать?** список функций и описание тестируемой системы и её компонент в отдельности

**Как будете тестировать?** - стратегия тестирования, а именно: **виды тестирования** и их применение по отношению к объекту тестирования

**Когда будете тестировать?** - последовательность проведения работ: подготовка (Test Preparation), тестирование (Testing), анализ результатов (Test Result Analysis) в разрезе запланированных фаз разработки

**Критерии начала тестирования:** -готовность тестовой платформы (тестового стенда) , законченность разработки требуемого функционала , наличие всей необходимой документации

**Критерии окончания тестирования:**

результаты тестирования удовлетворяют критериям качества продукта:

- 1) *Introduction (Введение)*
- 2) *Test Items (Объекты тестирования)*
- 3) *Features To Be Tested (Функциональности для тестирования)*
- 4) *Features Not To Be Tested (Функциональности которые не будут тестироваться )*
- 5) *Approach (Стратегия тестирования (виды, подходы, методы))*
- 6) *Item Pass/Fail Criteria (Критерии успешности тестирования )*
- 7) *Suspension Criteria and Resumption Requirements (Критерии остановки и возобновления тестирования )*
- 8) *Test Deliverables (Тестовые результаты)*
- 9) *Environmental Needs (Тестовое окружение)*
- 10) *Responsibilities (Ответственность)*



## *Содержание отчета*

1. Цель работы.
2. Краткие теоретические сведения.
3. Сгенерированные тесты различных видов для выбранного объекта реального мира.
4. Тестовые активности для сформулированных задач.
5. Выводы по работе.

## *Контрольные вопросы*

1. Что такое тестирование?
2. Какие существуют типы тестов по покрытию? Дайте характеристику каждому.
3. Какие существуют тестовые активности? Дайте характеристику каждому.
4. Какие существуют типы тестов знанию кода? Дайте характеристику каждому.
5. Какие существуют типы тестов по степени автоматизации? Дайте характеристику каждому.
6. Какие существуют типы тестов по изолированности компонентов? Дайте характеристику каждому.
7. Какие существуют типы тестов по подготовленности? Дайте характеристику каждому.
8. Какие существуют типы тестов по месту и времени проведения? Дайте характеристику каждому.
9. Какие существуют типы тестов по объекту тестирования? Дайте характеристику каждому.
10. Какие существуют типы функциональных тестов? Дайте характеристику каждому.
11. Какие существуют типы нефункциональных тестов? Дайте характеристику каждому.
12. Какие этапы составляют процесс тестирования?
13. Что происходит на этапе изучения и анализа предмета тестирования?
14. Что происходит на этапе планирования тестирования?
15. Что происходит на этапе исполнения тестирования?
16. Какие типы тестов выполняют для первой поставки программного продукта?
17. Какие типы тестов выполняют для последующих поставок программного продукта?