

**Министерство науки и высшего образования РФ
ФГБОУ ВО «Ульяновский государственный университет»
Факультет математики, информационных и авиационных технологий**

Смагин А.А.

**МЕТОДИЧЕСКИЕ УКАЗАНИЯ
ДЛЯ САМОСТОЯТЕЛЬНОЙ РАБОТЫ СТУДЕНТОВ
ПО ДИСЦИПЛИНЕ**

«Проектирование пользовательского интерфейса»

для студентов бакалавриата по направлению 11.03.02 - "Инфокоммуникационные технологии и системы связи"

Ульяновск, 2019

Методические указания для самостоятельной работы студентов по дисциплине «Проектирование пользовательского интерфейса» для студентов бакалавриата по направлению 11.03.02 - "Инфокоммуникационные технологии и системы связи" / составитель: Смагин А.А. – Ульяновск: УлГУ, 2019.

Настоящие методические указания предназначены для студентов бакалавриата направления 11.03.02 - "Инфокоммуникационные технологии и системы связи". В работе приведены литература по дисциплине, задание и материалы для выполнения лабораторных работ и методические указания для самостоятельной работы студентов.

Они будут полезны при подготовке к лабораторным занятиям и к зачету по данной дисциплине.

*Рекомендованы к введению в образовательный процесс Ученым советом Факультета математики, информационных и авиационных технологий УлГУ
(протокол № 3/19 от 19 апреля 2019 г.).*

Настоящие методические указания направлены на получение студентами теоретических знаний и практических навыков по современным технологиям создания пользовательских интерфейсов программного обеспечения и включает основные разделы знаний, необходимые для специалистов по обеспечению качества программных продуктов:

- классификация видов пользовательских интерфейсов;
- планирование тестирования пользовательских интерфейсов;
- разработка и тестирование требований к программному обеспечению;
- разработка интерфейсов;
- поиск и описание дефектов;
- оценка качества и документирование результатов тестирования пользовательских интерфейсов.

Подробно рассмотрены особенности юзабилити-тестирования, направленного на повышение эффективности человеко-компьютерного взаимодействия посредством разработки эргономичных интерфейсов.

Указания содержат базовые проверки графического интерфейса пользователя и функциональности программного обеспечения, примеры рабочей и отчетной тестовой документации, поясняющие иллюстрации.

1. ЛИТЕРАТУРА ДЛЯ ИЗУЧЕНИЯ ДИСЦИПЛИНЫ

1. Проскуряков, А. В. Качество и тестирование программного обеспечения. Метрология программного обеспечения : учебное пособие / А. В. Проскуряков. — Ростов-на-Дону, Таганрог : Издательство Южного федерального университета, 2022. — 197 с. — ISBN 978-5-9275-4044-0. — Текст : электронный // Цифровой образовательный ресурс IPR SMART : [сайт]. — URL: <https://www.iprbookshop.ru/125702.html>
2. Компаниец, В. С. Проектирование и юзабилити-исследование пользовательских интерфейсов : учебное пособие / В. С. Компаниец, А. Е. Лызь. — Ростов-на-Дону, Таганрог : Издательство Южного федерального университета, 2020. — 107 с. — ISBN 978-5-9275-3637-5. — Текст : электронный // Цифровой образовательный ресурс IPR SMART : [сайт]. — URL: <https://www.iprbookshop.ru/115528.html>
3. Барнум, К. М. Основы юзабилити-тестирования / К. М. Барнум ; перевод Д. А. Беликов. — Москва : ДМК Пресс, 2022. — 408 с. — ISBN 978-5-97060-960-6. — Текст : электронный // Цифровой образовательный ресурс IPR SMART : [сайт]. — URL: <https://www.iprbookshop.ru/126251.html>
4. Кохави, Р. Доверительное А/В-тестирование : практическое руководство по контролируемым экспериментам / Р. Кохави, Д. Тан, Я. Сюй ; перевод В. С. Яценков. — Москва : ДМК Пресс, 2021. — 297 с. — ISBN 978-5-97060-913-2. — Текст : электронный // Цифровой образовательный ресурс IPR SMART : [сайт]. — URL: <https://www.iprbookshop.ru/125112.html>
5. Плаксин, М. А. Тестирование и отладка программ для профессионалов будущих и настоящих / М. А. Плаксин. — 4-е изд. — Москва : Лаборатория знаний, 2020. — 168 с. — ISBN 978-5-00101-810-0. — Текст : электронный // Цифровой образовательный ресурс IPR SMART : [сайт]. — URL: <https://www.iprbookshop.ru/89029.html>
6. Карпович, Е. Е. Методы тестирования и отладки программного обеспечения : учебник / Е. Е. Карпович. — Москва : Издательский Дом МИСиС, 2020. — 136 с. — ISBN 978-5-907226-64-7. — Текст : электронный // Цифровой образовательный ресурс IPR SMART : [сайт]. — URL: <https://www.iprbookshop.ru/106722.html>
7. Гэртнер, М. ATDD - разработка программного обеспечения через приемочные тесты / М.

Гэртнер. Пер. с англ. А. А. Слинкин. - Москва : ДМК Пресс, 2016. - 232 с. - ISBN 978-5-97060-418-2. - Текст : электронный // ЭБС "Консультант студента" : [сайт]. - URL : <https://www.studentlibrary.ru/book/ISBN9785970604182.html>

2. СОДЕРЖАНИЕ ДИСЦИПЛИНЫ

Тема 1. Задачи и цели создания пользовательского интерфейса

Интерфейс. Базовые определения. Основные определения и виды программного пользовательского интерфейса. Пользовательский интерфейс. Функции. Структура.

Тема 2. Функциональное тестирование пользовательских интерфейсов

Анализ требований к пользовательскому интерфейсу. Разработка тест-требований и тест-планов для проверки пользовательского интерфейса. Выполнение тестовых примеров и сбор информации о выполнении тестов. Определение полноты покрытия пользовательского интерфейса требованиями. Составление отчетов о проблемах в случае несовпадения поведения системы и требований либо в случае отсутствия требований на отдельные интерфейсные элементы.

Тема 3. Тестирование удобства использования пользовательских интерфейсов

Исследовательское. Оценочное. Валидационное. Сравнительное. Наблюдаемость состояния системы. Соотнесение с реальным миром. Пользовательское управление и свобода действий. Целостность и стандарты. Помощь пользователям в распознавании, диагностике и устранении ошибок. Предотвращение ошибок. Распознавание, а не вспоминание. Гибкость и эффективность использования. Эстетичный и минимально необходимый дизайн. Помощь и документация.

Тема 4. Основные методы юзабилити-тестирования пользовательских интерфейсов

Основные понятия юзабилити. Юзабилити-тестирование. Полное и промежуточное тестирование. Анализ рабочих заданий. Сегментация пользовательской аудитории. Персонажи. Оценка производительности. Конструктивное взаимодействие. Фокус-группы. Метод карточной сортировки/ Экспертная оценка. Эвристическая оценка. Макетирование (прототипирование). Бумажное прототипирование.

Тема 5. Автоматизация тестирования пользовательского интерфейса

Понятие автоматизированного тестирования и его значение для разработки программного обеспечения. Уровни автоматизации. Место тестирования пользовательского интерфейса в общем процессе автоматизированного тестирования. Инструменты для автоматизации тестирования пользовательского интерфейса. Существующие подходы к автоматизации. Паттерн Page Object.

3. ПЕРЕЧЕНЬ ВОПРОСОВ К ЗАЧЕТУ

1. Интерфейс. Базовые определения.
2. Основные определения и виды программного пользовательского интерфейса. Пользовательский интерфейс. Функции. Структура.
3. Анализ требований к пользовательскому интерфейсу.
4. Разработка тест-требований и тест-планов для проверки пользовательского интерфейса.
5. Выполнение тестовых примеров и сбор информации о выполнении тестов.
6. Определение полноты покрытия пользовательского интерфейса требованиями.
7. Исследовательское тестирование.
8. Оценочное тестирование.
9. Валидационное тестирование.

10. Сравнительное тестирование.
11. Наблюдаемость состояния системы.
12. Соотнесение с реальным миром.
13. Пользовательское управление и свобода действий.
14. Целостность и стандарты.
15. Помощь пользователям в распознавании, диагностике и устранении ошибок.
16. Предотвращение ошибок.
17. Гибкость и эффективность использования.
18. Помощь и документация.
19. Основные понятия юзабилити.
20. Юзабилити-тестирование.
21. Полное и промежуточное тестирование.
22. Анализ рабочих заданий.
23. Сегментация пользовательской аудитории. Персонажи.
24. Оценка производительности.
25. Конструктивное взаимодействие. Фокус-группы.
26. Метод карточной сортировки.
27. Экспертная оценка.
28. Эвристическая оценка.
29. Макетирование (прототипирование).
30. Бумажное прототипирование.
31. Понятие автоматизированного тестирования и его значение для разработки программного обеспечения.
32. Уровни автоматизации.
33. Место тестирования пользовательского интерфейса в общем процессе автоматизированного тестирования.
34. Инструменты для автоматизации тестирования пользовательского интерфейса.
35. Существующие подходы к автоматизации. Паттерн Page Object.

4. ТЕСТОВЫЕ ЗАДАНИЯ ДЛЯ САМОСТОЯТЕЛЬНОЙ РАБОТЫ ПО ТЕОРЕТИЧЕСКОЙ ЧАСТИ ДИСЦИПЛИНЫ

Формулировка вопроса	Ответы
1. Какой из перечисленных пользовательских интерфейсов используется в ОС Windows?	<ul style="list-style-type: none"> • Командный • Голосовой • WIMP • SILK
2. Как называется совокупность средств и правил взаимодействия человека и компьютера?	<ul style="list-style-type: none"> • приложение • пользовательский интерфейс • объектный интерфейс
3. Какая разновидность пользовательского интерфейса появилась позже всех?	<ul style="list-style-type: none"> • Графический • Командный • SILK
4. Какой вид пользовательского интерфейса сейчас наиболее распространён?	<ul style="list-style-type: none"> • WIMP-интерфейс • командный • биометрический

5. Основными элементами графического интерфейса являются ...	<ul style="list-style-type: none"> • окна • файлы • команды
6. Командные кнопки для выполнения наиболее часто встречающихся операций располагаются...	<ul style="list-style-type: none"> • в адресной строке • в строке заголовка • на панели инструментов
7. Какой тип интерфейса позволяет проводить идентификацию пользователя по отпечаткам пальцев?	<ul style="list-style-type: none"> • интерфейс командной строки • на основе биометрической технологии • речевой интерфейс
8. Вид интерфейса, в котором взаимодействие осуществляется через набор команд, вводимых с клавиатуры – это	<ul style="list-style-type: none"> • Графический интерфейс • Голосовой интерфейс • Командный интерфейс
9. Выберите фазы через которые проходит функциональное тестирование пользовательских интерфейсов	<ul style="list-style-type: none"> • анализ требований к пользовательскому интерфейсу; • разработка тест-требований и тест-планов для проверки пользовательского интерфейса; • проектирование интерфейса • выполнение тестовых примеров и сбор информации о выполнении тестов;
10. Что из перечисленного является примером тестового непригодного требования?	<ul style="list-style-type: none"> • Пользовательский интерфейс должен быть интуитивно понятным. • Пользовательский интерфейс должен быть реализован так, чтобы функции системы были доступны при помощи не более чем 5 щелчков мыши по интерфейсным элементам.
11. Что изучает UI-тестирование	<ul style="list-style-type: none"> • Визуальный дизайн • Функциональность • Юзабилити • Производительность • Согласованность
12. Качественными данными в юзабилити-теста являются -	<ul style="list-style-type: none"> • числовые значения, которые пользователь дает нам в рамках теста; • ответы пользователей на открытые вопросы, например, когда пользователь отвечает на вопрос: «Что вы могли бы добавить в это приложение?»
13. К основным проверкам при тестировании UI относятся	<ul style="list-style-type: none"> • Расположение, размер, цвет, ширина, длина элементов; возможность ввода букв или цифр. • Реализуется ли функционал приложения с помощью графических элементов.

	<ul style="list-style-type: none"> • Проверка на то, реализовано ли данное решение правильно с точки зрения кодовой базы. • Читабелен ли использованный шрифт.
14. User Experience (пользовательский опыт) – это ...	<ul style="list-style-type: none"> • ощущение, испытываемое пользователем во время использования цифрового продукта • инструмент, позволяющий осуществлять интеракцию «пользователь — веб-ресурс»
15. User interface — это	<ul style="list-style-type: none"> • ощущение, испытываемое пользователем во время использования цифрового продукта • инструмент, позволяющий осуществлять интеракцию «пользователь — веб-ресурс»
16. GUI тестирование включает:	<ul style="list-style-type: none"> • Тестирование пользовательского интерфейса (UI) • Тестирование удобства использования (Usability Testing) • Compatibility testing (тестирование совместимости) • Всё из вышеперечисленного
17. Что относится к Тестированию пользовательского интерфейса (UI)	<ul style="list-style-type: none"> • навигация; • логичность. • цвета, графика, оформление; • содержание выводимой информации; • поведение курсора и горячие клавиши; • визуальное оформление; • отображение различного количества данных (нет данных, минимальное и максимальное количество); • изменение размеров окна или разрешения экрана.
18. Что относится к Тестированию удобства использования (Usability Testing)	<ul style="list-style-type: none"> • навигация; • логичность. • цвета, графика, оформление; • содержание выводимой информации; • поведение курсора и горячие клавиши; • визуальное оформление; • отображение различного количества данных (нет данных, минимальное и максимальное количество); • изменение размеров окна или разрешения экрана.
19. Тестирование пользовательского интерфейса (UI) – это ...	<ul style="list-style-type: none"> • тестирование, выполняемое путем взаимодействия с системой через графический интерфейс пользователя

	<ul style="list-style-type: none"> • процесс тестирования для определения возможности взаимодействия программного продукта, проверка работоспособности приложения в различных средах (браузеры и их версии, операционные системы, их типы, версии и разрядность) • тестирование с целью определения степени понятности, легкости в изучении и использовании, привлекательности программного продукта для пользователя при условии использования в заданных условиях эксплуатации
<p>20. Тестирование удобства использования (Usability Testing) – это ...</p>	<ul style="list-style-type: none"> • тестирование, выполняемое путем взаимодействия с системой через графический интерфейс пользователя • процесс тестирования для определения возможности взаимодействия программного продукта, проверка работоспособности приложения в различных средах (браузеры и их версии, операционные системы, их типы, версии и разрядность) • тестирование с целью определения степени понятности, легкости в изучении и использовании, привлекательности программного продукта для пользователя при условии использования в заданных условиях эксплуатации
<p>21. Compatibility testing (тестирование совместимости) - это</p>	<ul style="list-style-type: none"> • тестирование, выполняемое путем взаимодействия с системой через графический интерфейс пользователя • процесс тестирования для определения возможности взаимодействия программного продукта, проверка работоспособности приложения в различных средах (браузеры и их версии, операционные системы, их типы, версии и разрядность) • тестирование с целью определения степени понятности, легкости в изучении и использовании, привлекательности программного продукта для пользователя при условии использования в заданных условиях эксплуатации
<p>22. Что важно помнить при тестировании приложений с использованием метода тестирования «черный ящик»?</p>	<ul style="list-style-type: none"> • Совместимость приложения с несколькими устройствами

	<ul style="list-style-type: none"> • Правильное подключение и отключение устройства при смене IP или адреса порта • Возможность совершать звонки и отправлять сообщения на другие устройства • Совместимость с несколькими браузерами (Chrome, Firefox, Opera и т.д.) • Все вышеперечисленные варианты правильные
23. Выберите самую популярную среду для тестирования приложений Android.	<ul style="list-style-type: none"> • Webstorm • Visual Studio • PHPStorm • Android studio
24. Какое расширение у файлов Android?	<ul style="list-style-type: none"> • .ipa • .cs • .ts • .apk
25. Сколько типов средств автоматизации тестирования доступно для мобильных устройств?	<ul style="list-style-type: none"> • один, инструменты на основе объектов • один, инструменты на основе изображения • оба варианта правильны
26. xUnit – это ...	<ul style="list-style-type: none"> • Фреймворки модульного документирования • Фреймворки разработки через верификацию юнитов • Фреймворки модульного тестирования • Фреймворки юнит верификации
27. Преимущества тестирования в Eiffel	<ul style="list-style-type: none"> • Модульное тестирование, TDD • Eiffel BDD • Генерация тестов по модели • Autotest, тесты из отладчика
28. Что можно проверить и функциональным и модульным тестированием?	<ul style="list-style-type: none"> • Юнит-тест • Спецификацию • Прогрессию • Модуль • Регрессию
29. TDD - это	<ul style="list-style-type: none"> • Tradition-driven debugging • Test-driven debugging • Test-driven development • Turbo-driven development
30. Кто пишет модульные тесты	<ul style="list-style-type: none"> • Разработчик • Технический писатель • Тестирующий • Инженер по качеству

5. СПИСОК ЛАБОРАТОРНЫХ РАБОТ

Лабораторная работа №1

Тема: Проверка ссылок.

Цель работы: Проверка на сайте внутренних и внешних ссылок на наличие битых и отсутствующих страницы, наличие страницы 404, ошибка 500.

Задачи:

- научиться проверять актуальность ссылки с помощью онлайн-сервисов;
- научиться проверять ошибку 500;
- научиться проверять наличие страницы 404.

В лабораторной работе описываются способы проверки на сайте внутренних и внешних ссылок с помощью сторонних сервисов, а также описание ошибок.

Внешний вид лабораторной работы представлен на рисунке 1.

1. Проверка ссылок

Лабораторные работы

Лабораторная работа 1 «Проверка ссылок»

Цель работы. Проверка на сайте внутренних и внешних ссылок на наличие битых и отсутствующих страницы, наличие страницы 404, ошибка 500.

Задачи.

- Научиться проверять актуальность ссылки с помощью онлайн-сервисов;
- Научиться проверять ошибку 500;
- Научиться проверять наличие страницы 404.

Краткие теоретические сведения

БИТЫЕ ССЫЛКИ

Битая ссылка — ссылка на несуществующую страницу, документ, изображение. Переходя по битой ссылке пользователь, в лучшем случае, видит сообщение о том, что страница не найдена. В худшем случае он видит служебную страницу сервера.

Чтобы найти битые ссылки можно пройтись по всему сайту и понажимать по всем увиденным ссылкам. Это неплохой метод для сайта из пяти страниц, но что делать, когда количество страниц переваливает за несколько сотен, а то и тысяч? Ручная проверка явно не эффективна.

Существует несколько автоматизированных решений:

- Инструментарий вебмастера Яндекс и Google;
- Разнообразное программное обеспечение;
- Онлайн-сервисы.

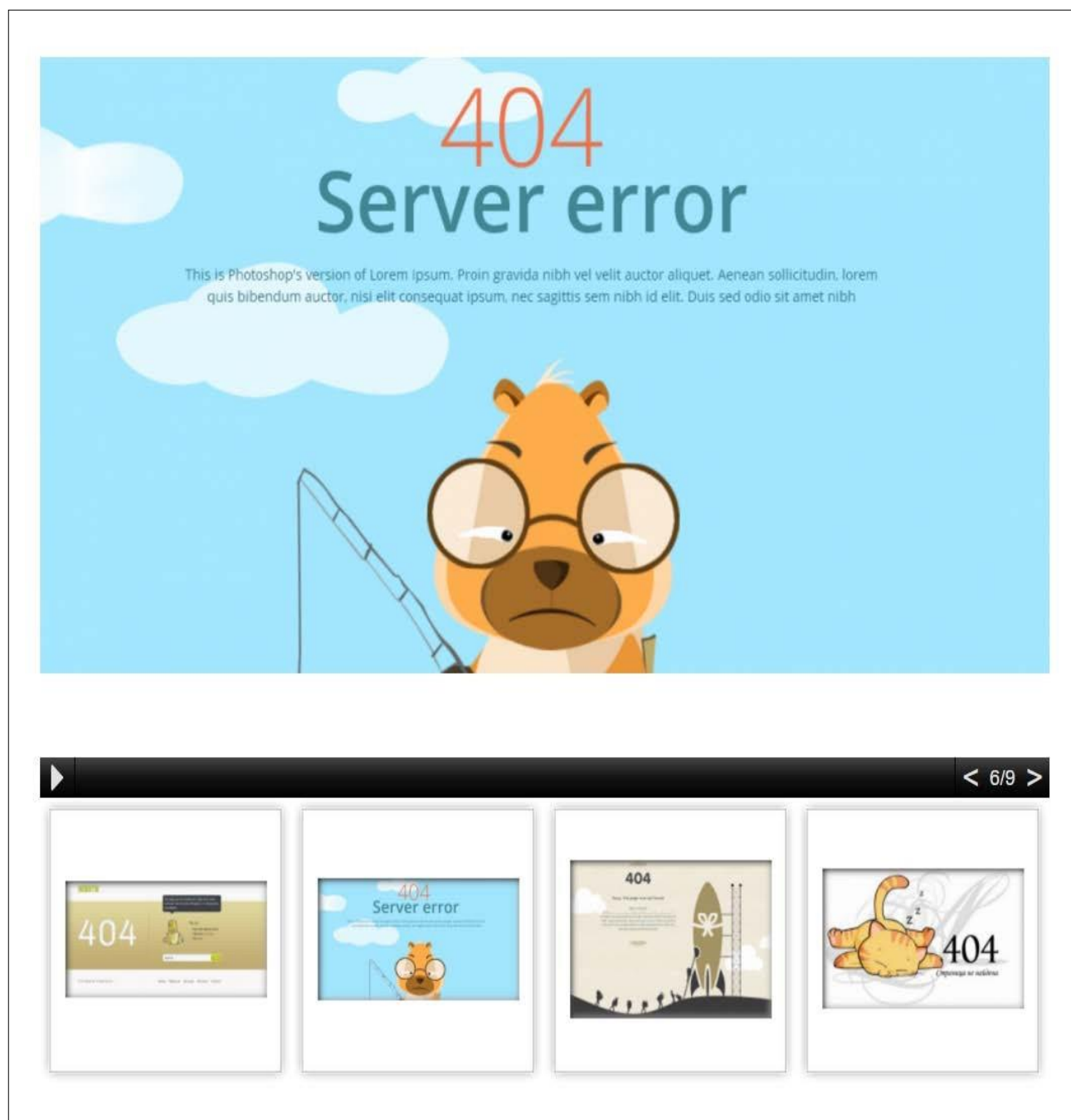
Задание:

1. Для онлайн проверки сайта на битые ссылки перейдем на сервер <http://www.brokenlinkcheck.com>. Он

Рисунок 1 – Лабораторная работа №1 «Проверка ссылок»

В процессе выполнения студенты анализируют сайт УлГУ на битые ссылки с использованием сторонних сервисов и рисуют страницу ошибки 404 и 500.

В лабораторной работе содержится галерея интересных примеров страниц 404 (рисунок 2).



Лабораторная работа №2

Тема: Тестирование юзабилити.

Цель работы: Проанализировать дизайн сайта на соответствие различным критериям.

Задачи:

- проверка соответствия логотипа на корпоративные цвета, цветовые линейки;
- принципы юзабилити сайта.

В лабораторной работе описываются способы тестирования юзабилити интерфейса сайта. Внешний вид лабораторной работы представлен на рисунке 3.

2. Тестирование юзабилити

Лабораторные работы

Лабораторная работа 2 «Тестирование юзабилити»

Цель работы. Проанализировать дизайн сайта на соответствие различным критериям.

Задачи.

- Проверка соответствия логотипа на корпоративные цвета, цветовые линейки;
- Принципы юзабилити сайта.

Краткие теоретические сведения

ЮЗАБИЛИТИ

Под понятием юзабилити сайта подразумевается простое и удобное использование веб-ресурса посетителями.

Основные проблемы юзабилити сайтов, с которыми сталкиваются пользователи это:

- отсутствие хорошей навигации;
- сложная или запутанная структура;
- неудачное расположение элементов сайта;
- неуместное использование рекламы.

ВИЗУАЛЬНАЯ ИЕРАРХИЯ

Визуальная иерархия один из важнейших принципов эффективного дизайна. Фактически это порядок, в котором человеческий глаз воспринимает увиденное.



Рисунок 3 – Лабораторная работа №2 «Тестирование юзабилити»

В рамках лабораторной работы проверяется дизайна на соответствие различным критериям, таким как, соответствие принципам Usability:

1. Принцип «Визуальная иерархия».

Визуальная иерархия один из важнейших принципов эффективного дизайна. Фактически это порядок, в котором человеческий глаз воспринимает увиденное. Не зная ничего об этих кругах, вы можете легко распределить их по важности. Это и есть визуальная иерархия (рисунок 4).

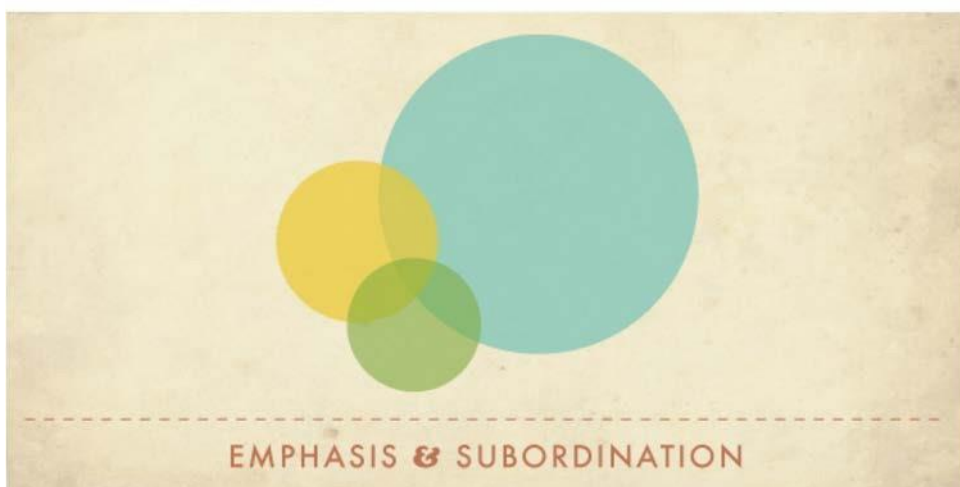


Рисунок 4 – Принцип визуальной иерархии

Визуальной иерархии элементов можно добиться не только с помощью размера. Например, разработчики сайта Amazon решили выделить кнопку «Добавить в корзину» цветом (рисунок 5).



Рисунок 5 – Пример выделения кнопки цветом

В лабораторной работе студенты должны оценить расположение и размеры значимых блоков.

2. Принцип «Золотое сечение».

Золотое сечение – это магическое число 1,618 (), которое делает все вещи пропорциональными и эстетически привлекательными (по крайней мере, считается, что это так). Вот пример того, как выглядит золотое сечение (рисунок 6).

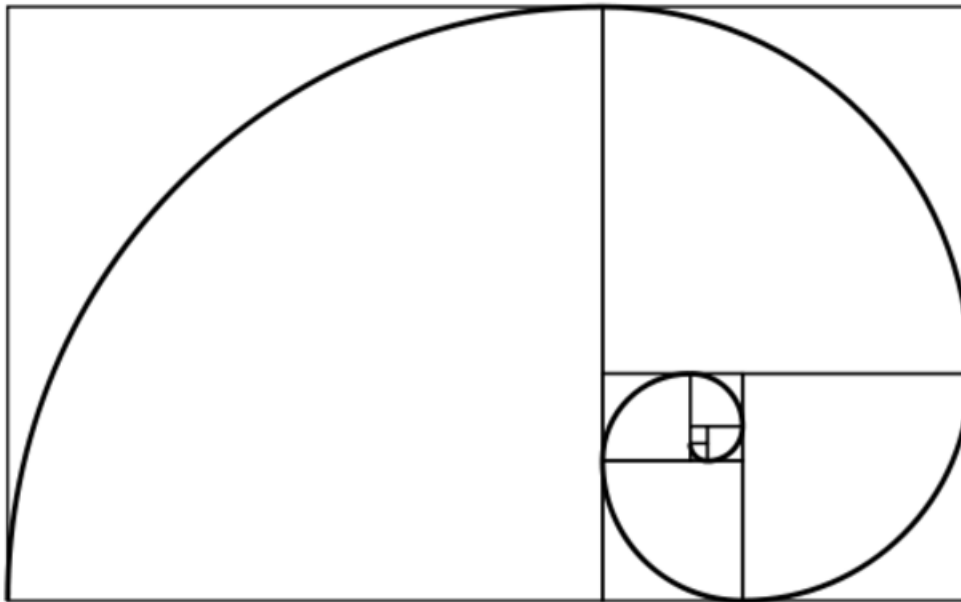


Рисунок 6 – Принцип золотого сечения

В лабораторной работе студенты должны оценить и применить золотое сечение при проверке сайта. Вот, к примеру, сайт (рисунок 7):

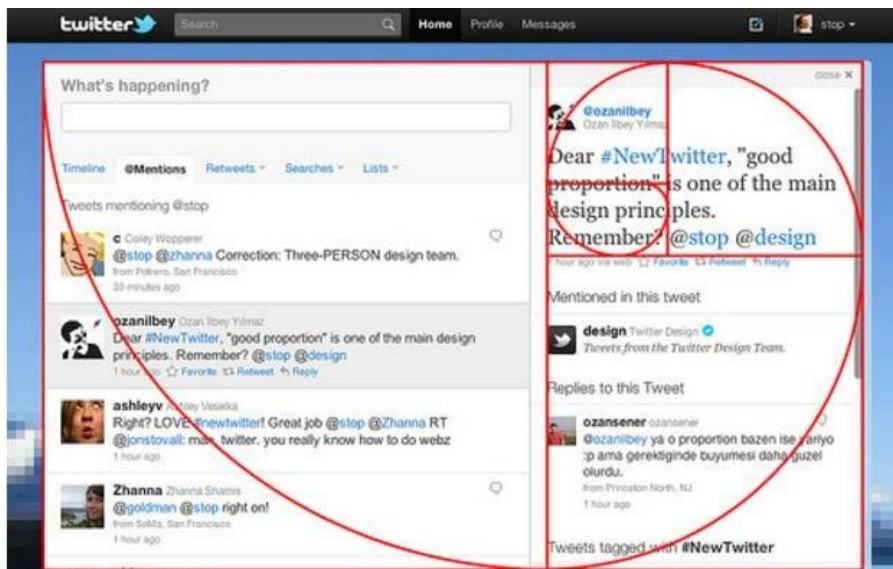


Рисунок 7 – Пример применения золотого сечения

3. Принцип «Баланса».

Баланс — это равновесие взаимодействующих или противоположных сил в композиции. В такой композиции не возникает ощущения, что какая-то ее часть доминирует над всем остальным. Баланса можно достичь с помощью правильного размещения объектов, размеров объекта и по цвету. Баланс может быть

симметричным (верхний рисунок), асимметричным (нижний рисунок), радиальным (объекты расположены кругом и расходятся из одной точки) (рисунок 8).

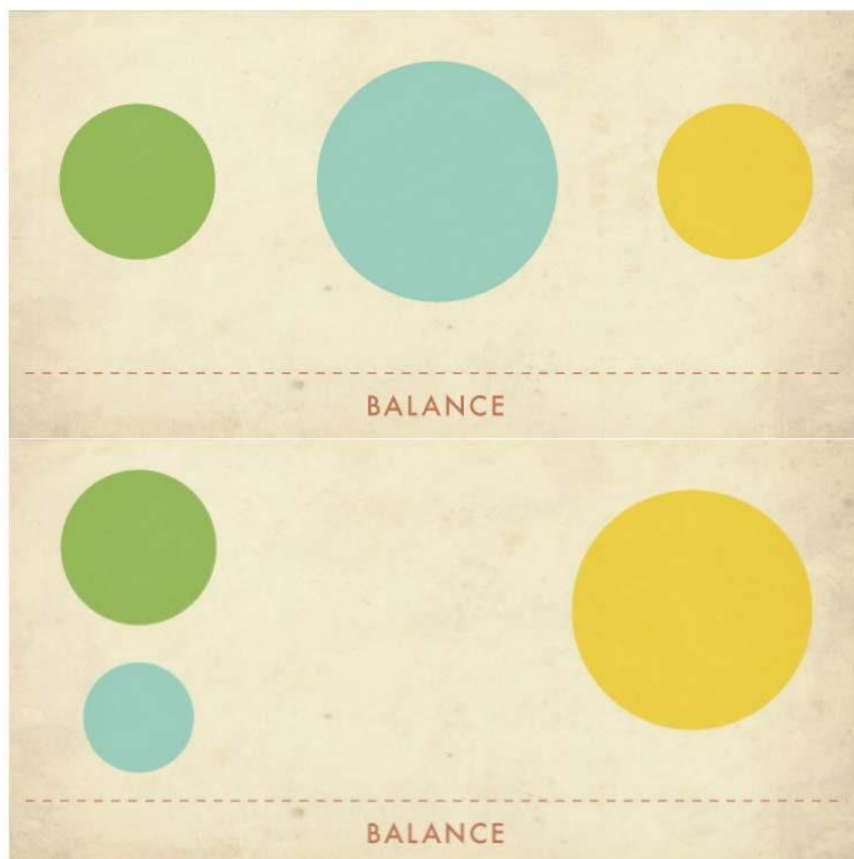


Рисунок 8 – Принцип баланса

В лабораторной работе оцениваются расположение элементов в разрезе равновесия (рисунок 9).

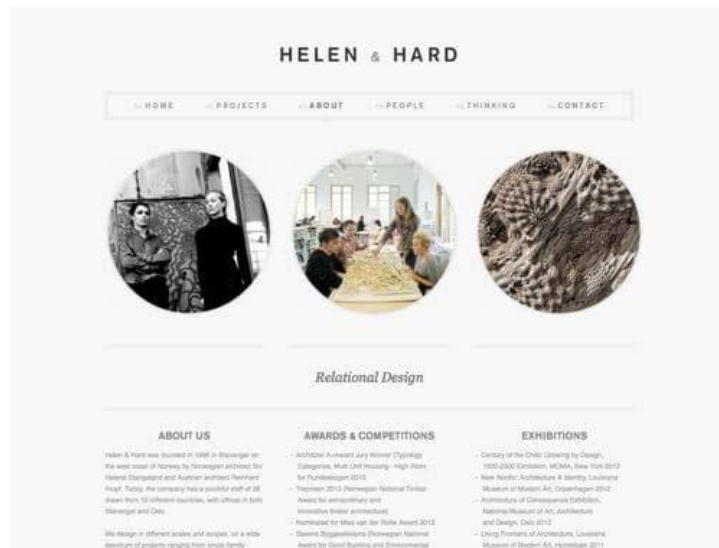


Рисунок 9 – Пример применения принципа баланса

4. Принцип «Контраста».

Контраст — это взаимодействие противоположных элементов композиции, таких как цвет, размер, текстура и т.д. Примеры контраста: большой и маленький, шероховатый и гладкий, толстый и тонкий, черный и белый (рисунок 10).

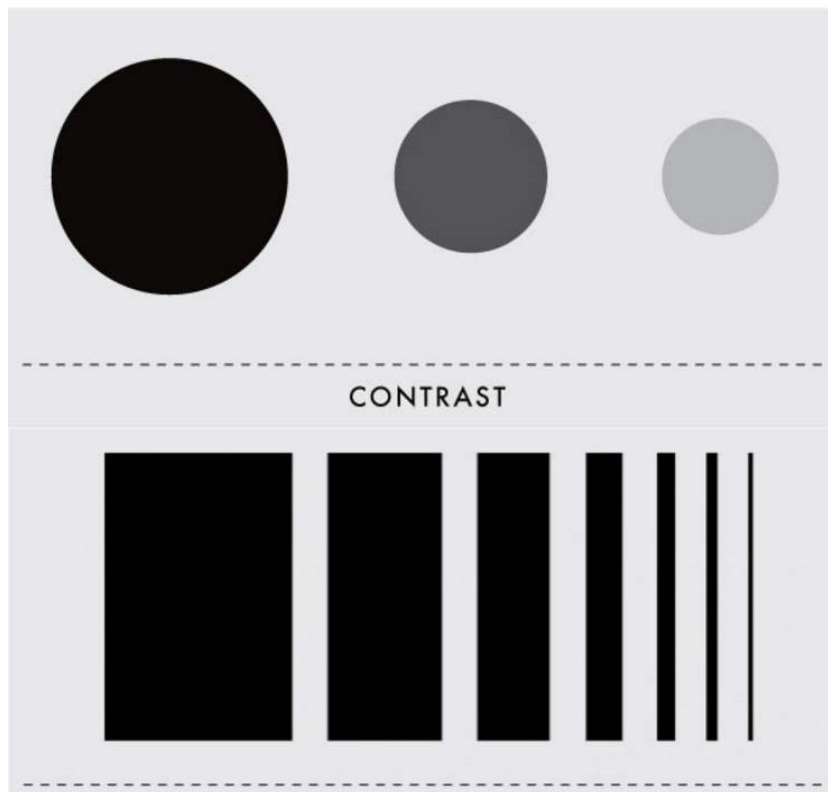


Рисунок 10 – Принцип контраста

В лабораторной работе осуществляется поиск в интернете контрастных сайтов

и анализ их страниц (рисунок 11).



Рисунок 11 – Пример контраста на сайте

Лабораторная работа №3

Тема: Тестирование бизнес-логики.

Цель работы: Ознакомиться с тестированием бизнес-логики и интерфейса, выполнить проверку валидности форм, работоспособность капчи и регистрацию на сайте.

Задачи:

- проверка валидности форм;
- ввод некорректных символов в форму;
- проверка работоспособности капчи;
- проверка поступления сообщения о регистрации на почту.

В лабораторной работе описываются способы тестирования заполняемых форм на сайтах и проверка работы процессов.

Внешний вид лабораторной работы представлен на рисунке 12.

3. Тестирование бизнес-логики

Лабораторные работы

Лабораторная работа 3 «Тестирование бизнес-логики»

Цель работы. Ознакомиться с тестированием бизнес-логики и интерфейса, выполнить проверку валидности форм, работоспособность капчи и регистрацию на сайте.

Задачи.

- Проверка валидности форм;
- Ввод некорректных символов в форму;
- Проверка работоспособности капчи;
- Проверка поступления сообщения о регистрации на почту.

Краткие теоретические сведения

БИЗНЕС-ЛОГИКА

Бизнес-логика - это описание отношений, поведения между элементами предметной области, процессов, происходящих в той сфере, которая реализуется в приложении, и правил, по которым эти процессы происходят.

В первую очередь в приложении реализуются уже на языке программирования основные понятия системы: объекты, классы или модели, описывающие сущности предметной области. А затем уже реализуется бизнес-логика, то есть процессы и правила.

1. Путь (URL)

Положительный эффект: после обработки данных формы пользователь должен попадать на некую "информативную страницу", т.е., например: пользователь хочет изменить количество заказанных товаров в корзине, он меняет число в соответствующем поле, кликает "Пересчитать", и должен оказаться снова на странице корзины. Проще всего достичь этого эффекта можно так: указать в атрибуте action тега form URL самой же страницы, на которой расположена форма.

Рисунок 12 – Лабораторная работа №3 «Тестирование бизнес-логики»

Формы в общем случае делятся на два основных вида:

- однооконные формы – полностью (со всеми своими полями) расположены на одной странице;
- пошаговые формы – новые поля появляются (после загрузки новой страницы или в рамках старой с использованием JavaScript/AJAX) по мере заполнения уже показанных.

В рамках лабораторной работы проверяются заполняемые формы на соответствие различным критериям:

1. Путь (URL)

Положительный эффект: после обработки данных формы пользователь должен попадать на некую «информативную страницу», т.е., например:

пользователь хочет изменить количество заказанных товаров в корзине, он меняет

число в соответствующем поле, кликает «Пересчитать», и должен оказаться снова на странице корзины. Проще всего достичь этого эффекта можно так: указать в атрибуте action тега form URL самой же страницы, на которой расположена форма.

Отрицательный эффект: вместо страницы корзины появляется надпись «Спасибо, операция выполнена успешно». В лучшем случае после этого происходит редирект на страницу корзины. В худшем – пользователю приходится нажимать в браузере Back и... видеть подхваченную из кэша страницу, на которой остались старые данные.

2. Месторасположение.

Положительный эффект: если форма расположена достаточно низко, сделана автоматическая прокрутка к форме.

Отрицательный эффект: страница с формой или результатами выполнения операции открывается в состоянии «вверху окна браузера – верх страницы». Форма или некоторая информация о выполненной операции видна частично или не видна вообще.

3. Ошибочные ситуации.

Положительный эффект: в случае, если пользователь ввёл некоторые данные некорректно (или не ввёл вообще), форма должна быть показана заново, и при этом все введённые данные (за исключением паролей и полей капчи) должны сохранять свои, в удобном для восприятия месте (чаще всего – сразу над формой) должна быть указана причина неудачи отправки данных.

Отрицательный эффект: Значения полей «обнуляются», Причина неудачи отправки данных не объяснена, ошибочно заполненные поля не указаны.

4. Значения полей и спецсимволы.

Положительный эффект: через поле, теоретически, могут быть введены любые символы. Из этого следует необходимость тщательной фильтрации данных перед передачей их в БД или иной приёмник.

Отрицательный эффект: пользователь заполняет поле неверно и проходит

дальше.

Внешний вид отчета по лабораторной работе №3 «Тестирование бизнес-логики» представлен на рисунке 13.

Отчет по лабораторной работе 3 «Тестирование бизнес-логики»

Студент Ф.И.О.

Группа номер группы

1. Посмотрите, исправно ли работают формы, вставьте скриншот.
Скриншот результата:
Заполните поля, в которых появились сообщения об ошибке, а поля в которых ее нет оставьте пустыми. Проверьте результат, вставьте скриншот.
Скриншот результата:
2. Введите некорректные символы в поля заполнения формы, проверьте результат, вставьте скриншот.
Скриншот результата:
3. Проверьте работоспособность капчи, обновите, введите неправильно. Вставьте скриншот каждого действия.
Скриншот результата:
4. Проверьте, поступило ли сообщение о подтверждении регистрации на почтовый ящик. Вставьте скриншот.
Скриншот результата:

Результат отправьте преподавателю.

Рисунок 13 – Отчет по лабораторной работе 3 «Тестирование бизнес-логики»

Лабораторная работа №4

Тема: Тестирование навигации.

Цель работы: Ознакомиться с тестированием навигации, понять способы перехода по сайту.

Задачи:

- посчитать количество переходов до определенного места сайта;
- протестировать поиск по сайту;
- проверить навигацию кнопки «наверх»;
- восстановить путь по адресной строке.

В лабораторной работе описываются способы тестирования навигации, процедуры перемещения пользователей из одного места в другое.

Внешний вид лабораторной работы представлен на рисунке 14.

4. Тестирование навигации

Лабораторные работы

Лабораторная работа 4 «Тестирование навигации»

Цель работы. Ознакомиться с тестированием навигации, понять способы перехода по сайту.

Задачи.

- Количество переходов до определенного места сайта;
- Тестирование поиска по сайту;
- Проверка кнопки «наверх»;
- Восстановление пути по адресной строке.

Краткие теоретические сведения

ТЕСТИРОВАНИЕ НАВИГАЦИИ

Помните, что наибольшее число посетителей любого сайта составляют неопытные пользователи Интернета. Им нужна возможность перемещаться по вашему сайту самым простым и быстрым способом.

Поэтому не пытайтесь поразить своих зрителей причудливой системой навигации. Напротив, создайте простую и функциональную систему. Подумайте, что более всего цените в любом Web-сайте вы лично. Вам нужно найти то, что вы ищете. То же самое нужно и вашим посетителям.

Процедуры перемещения посетителей из одного места сайта в другое ни в коем случае не должны быть испытанием для их терпения. Наоборот, такой переход должен вознаграждать посетителей ценной информацией и вызывать у них ощущения легкости и непринужденности, что побудит их задержаться на вашем сайте подольше - а не уходить из сайта с чувством разочарования и досады.

ДИНАМИЧЕСКАЯ НАВИГАЦИЯ

Навигация по странице должна подсказывать посетителю его следующее действие. К примеру, все мы привыкли, что при наведении указателя на элемент с видео, стрелочка меняется на указательный палец. Или при наведении курсора на текст, стрелка меняется на курсор-выделение.

Рисунок 14 – Лабораторная работа №4 «Тестирование навигации»

Лабораторная работа №5

Тема: Кроссбраузерное тестирование.

Цель работы: Ознакомиться с кроссбраузерным тестированием, провести ручное тестирование на нескольких браузерах.

Задачи:

- выбор браузеров для тестирования;
- тестирование сайта в различных браузерах.

В лабораторной работе описываются способы тестирования отображения

Web-приложения в различных браузерах, на примере самых популярных. Проверяется совместимость в цвете, шрифтах, расположении картинок и других элементов.

Для выполнения работы студенты должны скачать и установить различные версии браузеров или воспользоваться отдельным сервисом позволяющим поэкспериментировать со стилями для просмотра.

Внешний вид лабораторной работы представлен на рисунке 15.

5. Кроссбраузерное тестирование
Лабораторные работы

Лабораторная работа 5 «Кроссбраузерное тестирование»

Цель работы. Ознакомиться с кроссбраузерным тестированием, провести ручное тестирование на нескольких браузерах.

Задачи.

- Выбор браузеров для тестирования;
- Тестирование сайта в различных браузерах.

КРОССБРАУЗЕРНОЕ ТЕСТИРОВАНИЕ

Тестирование кроссбраузерности — вид тестирования, направленный на поддержку и правильное полное отображение программного продукта в разных браузерах, мобильных устройствах, планшетах, экранах различного размера.

Кроссбраузерное тестирование (cross-browser testing) — важный этап при разработке любой программы. Внешний вид сайта и его корректное отображение на любом современном устройстве играет определяющую роль для заказчика.

Cross-browser testing сайта начинается с выбора браузеров. Заказчик сам определяет, с какими именно веб-обозревателями будет работать его приложение. Но задача разработчика и тестировщика — подсказать клиенту, какой браузер будет главным, следует изучить статистику заходов подобных приложений, определить какими браузерами пользуется такая аудитория.

Как правило, рассматривают самые популярные браузеры: Google Chrome, Mozilla Firefox, Internet Explorer, Yandex Browser.

Основные моменты для тестирования: вёрстка (цвет, шрифты, расположение графических картинок и динамических элементов)

Рисунок 15 – Лабораторная работа №5 «Кроссбраузерное тестирование»

В разделе присутствуют сервисы для тестирования, такие как:

1. LoadImpact: (loadimpact.com) – сервис нагрузочного тестирования сайтов.
2. BrowserMob (browsermob.com) – сервис позволяющий проверить скорость загрузки целевого сайта из 4 разных мест и показывающий много подробных метрик загрузки для каждого из них. А основная задача BrowserMob — автоматическое

кросс-браузерное тестирование сайтов.

3. Alertra (alertra.com) – сервис мониторинга аптайма, на сайте которого можно бесплатно проверить доступность (и время ответа) сайта из 11 разных мест.

4. site24x7 (site24x7.com) – похожий на Alertra сервис, который использует 23 географически распределенные точки проверки.

5. You get signal (www.yougetsignal.com) – сервис позволяющий проверить открытые порты на любом хосте/IP. Помогает для проверки проброса портов через NAT-ы.

6. CSE HTML Validator (www.onlinewebcheck.com) – альтернатива известнейшему validator.w3.org, построенная на своем отдельном ядре анализа. Выгодно отличается очень подробными и понятными рекомендациями по исправлению ошибок и потенциальных проблем.

7. Built with: builtwith.com анализирует заданный сайт и выдает целое досье о нем: веб-сервер, CMS, какая служба статистики используется и так далее.