



Министерство образования Республики Беларусь

Учреждение образования
«Гомельский государственный технический
университет имени П. О. Сухого»

Институт повышения квалификации
и переподготовки

Кафедра «Профессиональная переподготовка»

В. А. Бельский

ИНФОРМАЦИОННАЯ АРХИТЕКТУРА И ЮЗАБИЛИТИ

ПРАКТИКУМ

**по выполнению лабораторных работ
по одноименному курсу для слушателей
специальности переподготовки**

**1-40 01 74 «Web-дизайн и компьютерная графика»
заочной формы обучения**

Гомель 2020

УДК 004.5(075.8)
ББК 32.811я73
Б44

*Рекомендовано кафедрой «Профессиональная переподготовка» ИПКиП
ГГТУ им. П. О. Сухого
(протокол № 4 от 20.12.2019 г.)*

Рецензент: проф. каф. «Информатика» ГГТУ им. П. О. Сухого
д-р техн. наук, проф. *В. П. Кудин*

Бельский, В. А.

Б44 Информационная архитектура и юзабилити : практикум по выполнению лаборатор. работ по одноим. курсу для слушателей специальности переподготовки 1-40 01 74 «Web-дизайн и компьютерная графика» заоч. формы обучения / В. А. Бельский. – Гомель : ГГТУ им. П. О. Сухого, 2020. – 40 с. – Систем. требования: PC не ниже Intel Celeron 300 МГц ; 32 Mb RAM ; свободное место на HDD 16 Mb ; Windows 98 и выше ; Adobe Acrobat Reader. – Режим доступа: <http://elib.gstu.by>. – Загл. с титул. экрана.

Изложены вопросы, связанные с тестированием различных программных продуктов. Основное внимание при этом уделяется тестированию юзабилити. Рассмотрены практические приемы расчетов различных юзабилити-метрик, применяемых для оценки качества интерфейса программного продукта.

**УДК 004.5(075.8)
ББК 32.811я73**

© Учреждение образования «Гомельский
государственный технический университет
имени П. О. Сухого», 2020

СОДЕРЖАНИЕ

Лабораторная работа № 1. Виды тестирования. Планирование тестирования	4
Лабораторная работа № 2. Тестирование юзабилити	15
Лабораторная работа № 3. Анализ качества интерфейсов с помощью юзабилити-метрик	26
Приложения	32

Лабораторная работа № 1. Виды тестирования. Планирование тестирования

Цель работы: изучить классификацию видов тестирования, практически закрепить эти знания путем 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 правильной работы 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, которые указаны 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 (АТ) 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 АТ.

Типы тестов по знанию кода

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 и т.д.).

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 – тестирование с целью определения степени понятности, легкости в изучении и использовании, привлекательности программного продукта для пользователя при условии использования в заданных условиях эксплуатации.

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:

- 1) какие функциональности предстоит протестировать,
- 2) как эти функциональности работают.

Планирование тестирования происходит на стадии разработки (кодирования) программного обеспечения. На стадии планирования тестирования перед тестировщиком 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 первой 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 багтрекинг-систему.

Для последующих вариантов композиции тестов могут быть следующими:

- Если не была добавлена новая функциональность, то: DV+MAT. Иными словами, 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 1.1).
 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 расширить рынки сбыта и просит осуществить поддержку для Великобритании (кроме уже существующей Беларуси).
 - 3.4. Заказчик хочет убедиться, что ПО держит нагрузку в 2000 пользователей.
- 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.

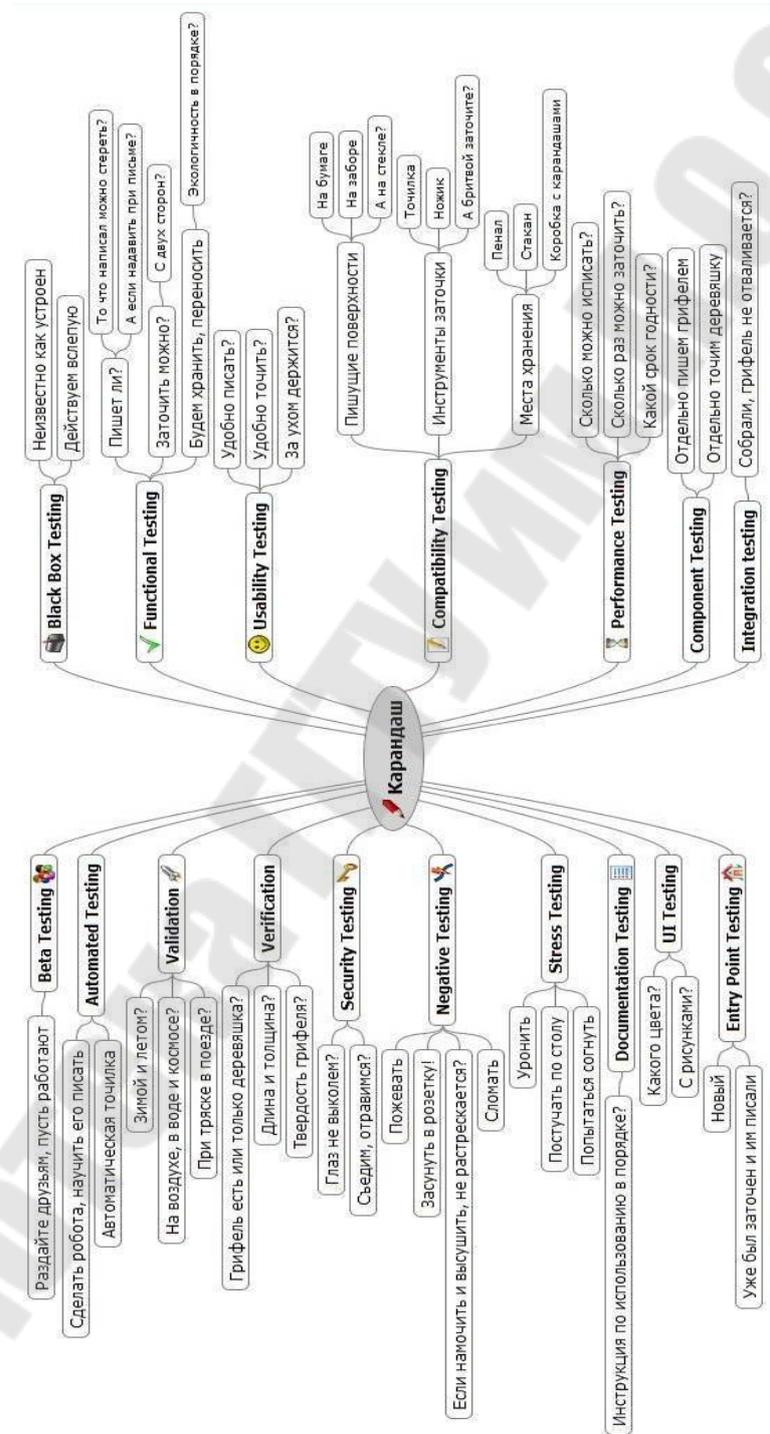


Рисунок 1.1 – Пример генерации тестов различных видов для объекта «Карандаш»

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 типу.
4. Какие существуют типы тестов по знанию кода? Дайте характеристику каждому типу.
5. Какие существуют типы тестов по степени автоматизации? Дайте характеристику каждому типу.
6. Какие существуют типы тестов по изолированности компонентов? Дайте характеристику каждому типу.
7. Какие существуют типы тестов по подготовленности? Дайте характеристику каждому типу.
8. Какие существуют типы тестов по месту и времени проведения? Дайте характеристику каждому типу.
9. Какие существуют типы тестов по объекту тестирования? Дайте характеристику каждому типу.
10. Какие существуют типы функциональных тестов? Дайте характеристику каждому типу.
11. Какие существуют типы нефункциональных тестов? Дайте характеристику каждому типу.
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 выполняются для первой поставки программного продукта?

Лабораторная работа № 2. Тестирование юзабилити

Цель работы: выполнить анализ целевой аудитории, составить карту эмпатии, разработать персонажей и карту сценария для проектирования систем, соответствующих принципам юзабилити.

Теоретические сведения

Юзабилити – степень, с которой продукт может быть использован определёнными пользователями при определённом контексте для достижения определённых целей с должной эффективностью, результативностью и удовлетворённостью.

Юзабилити отражает степень удобства использования программного продукта конечными пользователями. Так как взаимодействие пользователя и программного обеспечения осуществляется посредством пользовательских интерфейсов, то понятие юзабилити прежде всего относится к процессу разработки пользовательских интерфейсов.

Ключевые аспекты юзабилити – пользователь, контекст, цель. Для проектирования пользовательских интерфейсов, ориентированных на максимальное удобство пользователя, а также для последующего юзабилити-тестирования прежде всего необходимо определить целевую аудиторию. Во-вторых, важен контекст – ситуация, при которой пользователь обращается к программному продукту. При этом требуется не только определить контекст, но и воспроизвести его. В-третьих, важна цель (цель пользователя, цель бизнеса).

Первый этап. Исследование целевой аудитории

Целевая аудитория – группа пользователей, на которую сфокусировано содержание программного продукта. Целевые пользователи точно знают, в получении какой информации они заинтересованы и какой именно товар или услугу желают приобрести.

Выявление целевой аудитории поможет:

- понять, чем именно конкурирующие продукты привлекают ваших потенциальных пользователей;
- не создать «Франкенштейна»: продукт, в котором так много функций «для всех», что им не может пользоваться никто;

- сфокусироваться на главном, сэкономив средства на разработку;
- повысить эффективность работы пользователей и их удовлетворенность;
- обнаружить эффективные маркетинговые каналы для вашей компании.

Изучение целевой аудитории необходимо для проектирования взаимодействия в целом и корректировки элементов интерфейса. Это в итоге направлено на удержание текущих пользователей и привлечение потенциальных пользователей.

В итоговом документе, составленном по результатам исследования целевой аудитории, приводятся следующие данные:

- 1) описание аудитории – общий взгляд на целевую аудиторию;
- 2) структура аудитории – сегментирование целевой аудитории;
- 3) объем аудитории – количество людей в составе целевой аудиторией;
- 4) предпочтения – какие предпочтения свойственны целевой аудитории в целом и ее отдельным группам в частности;
- 5) факторы принятия решения о покупке – какие факторы влияют на принятие решения о покупке товара и какие моменты являются определяющими;
- 6) социально-демографические характеристики – каковы ключевые характеристики целевой аудитории: пол, возраст, социальный статус и т.п.;
- 7) тенденции – как будет изменяться объем, состав и предпочтения целевой аудитории в ближайшем будущем.

Подробные способы получения указанной информации приведены ниже.

При исследовании целевой аудитории на первом этапе необходимо составить ее **общий портрет**, учитывая разные характеристики:

Социально-демографические характеристики (основополагающая информация): пол, возраст, образование, уровень дохода, род занятий. Например, сайт для подростков 15–18 лет будет отличаться от сайта для пожилых людей в возрасте 60+ лет.

Психологические характеристики: стиль жизни, особенности личности, черты характера, жизненная позиция, система ценностей. Более ценная информация для проектирования, чем первая группа

критериев. Например, если известно, что целевая аудитория больше всего ценит время, можно спроектировать простой интерфейс и дать возможность получать не весь контент, а самое ценное для конкретной целевой группы, или даже дать инструменты персонализации каждого человека.

Поведенческие характеристики: повод для регистрации, искомые выгоды, частота посещаемости конкурентов, степень готовности к переходу на другой продукт, отношение к проекту (если он не новый) и т.д. Эта группа показателей одна из самых важных для проектирования. В тоже время, собрать эти данные будет очень сложно. Эта информация может быть у заказчика, если проектируется новая версия уже существующего проекта, у конкурента, или её нужно будет собирать по крупицам через опросы целевой аудитории или составление карт эмпатии.

Географические характеристики: страна, город, район. В общем случае это маловажный критерий, однако если стоит задача по проектированию национальных продуктов или продуктов с геолокацией, то важность этого критерия резко вырастает. Кроме того, если есть географическая привязка, это может повлиять на контент, о котором тоже нужно думать при проектировании.

Для получения данных, составляющих общий портрет целевой аудитории, используют различные **способы сбора информации:**

1. **Данные счетчика посетителей** (позволяют изучить все действия пользователей на сайте и конкретизировать распределение аудитории сайта по регионам, по времени и др.);
2. **Статистика запросов поисковых систем.** Оценить величину целевой аудитории можно по числу поисковых запросов. Подобные сервисы имеют, например, поисковые системы Яндекс (Яндекс статистика поиска), Рамблер (Adstat Rambler) и Google (Google Статистика поиска).
3. **Опросы аудитории** (анкетирование пользователей, использование опросной формы или регистрации на сайте);
4. **Системы аудита и традиционные опросы исследовательских компаний (агентств).**
5. **Анализ данных из социальных сетей и других публичных источников.**
6. **Составление карты эмпатии.**

Поведенческие характеристики целевой аудитории помогают понять привычки, мотивацию, круг интересов, проблемы, надежды и ожидания потребителя (как именно человек делает выбор, что влияет на решение купить продукт или отказаться от покупки, какой параметр является главным (качество товара, известная марка, стоимость, мнение друзей и т. п.)), для этого полезно составлять карту эмпатии.

Эмпатия – это психологический термин, который отражает способность понимать чувства и настроения других людей. Это не просто эмоциональное сопереживание, а умение поставить себя на место другого и увидеть мир его глазами.

Карта эмпатии – это метод исследования целевой аудитории. Из потребительских сегментов выбирают конкретных потребителей, с которыми проводится кропотливая работа: группа заполняет анкеты и проходит тестирование. При этом важно наблюдать за поведением каждого человека в группе и анализировать его слова и поступки, чтобы на основании всей этой информации составить полную картину. Главная задача карты эмпатии – составить подробный портрет типичного потребителя для конкретного продукта.

Карта эмпатии представляет собой диаграмму профиля потребителя, состоящую из шести сегментов (рисунок 2.1).

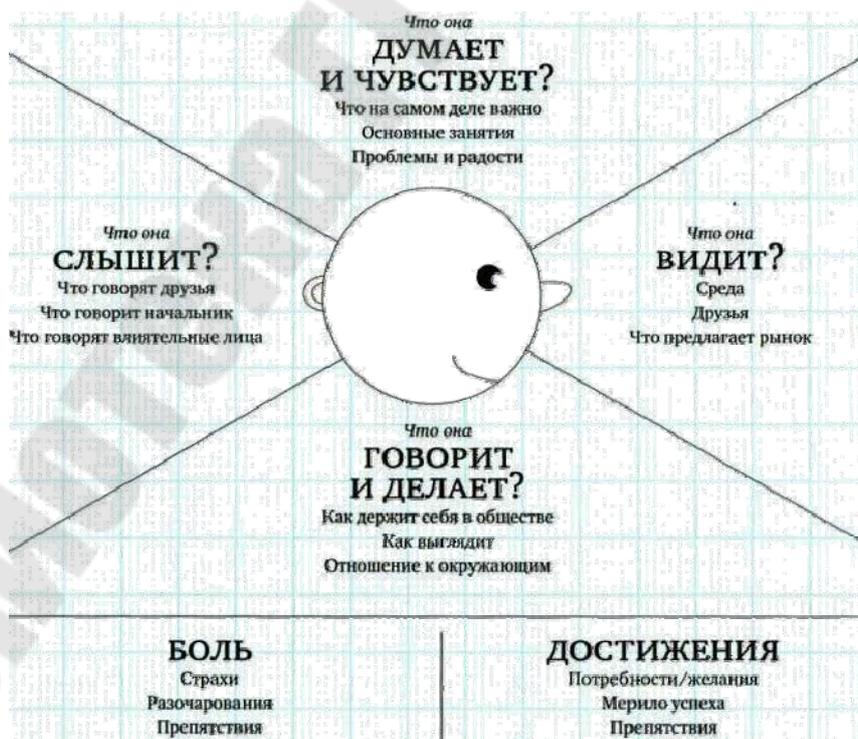


Рисунок 2.1 – Карта эмпатии

Правильное представление о целевой аудитории помогает адекватно позиционировать новый продукт, грамотно проводить его рекламную кампанию, а значит, повышать уровень продаж. Анализ карты эмпатии покажет, в чем ваш продукт не соответствует требованиям или ожиданиям целевой аудитории и как этот недостаток исправить.

Блоки «вижу» и «слышу» позволяют определить оптимальные каналы распространения информации о разрабатываемой системе. Особое внимание следует уделять последним двум блокам карты эмпатии: презентация продукта должна развеять все сомнения и тревоги, заключенные в блоке «болевые точки/боль», и максимально подчеркивать способность этого товара/услуги помочь в реализации целей из блока «успехи и стремления/достижения». Важно уделить внимание возможному конфликту между тем, что человек «говорит и делает» на публике, и тем, как он «думает и чувствует» на самом деле.

7. Разработка персонажей.

Для детализации и визуализации сведений о целевой аудитории описываются конкретные «персонажи», или «персоны» как типичные представители различных групп целевой аудитории (если она неоднородна).

Персонаж – это реалистичный собирательный образ пользователя, представляющий один сегмент целевой аудитории. Персонажи отличаются моделью поведения, целями, задачами, потребностями и ценностями.

Характеристики персонажа, кроме портрета, должны быть синтезированными: это означает, что они должны совмещать в себе все значимые характеристики представителя группы целевой аудитории, с добавлением специфических черт для придания персонажу реалистичности.

Обобщённый профиль представителя целевой аудитории, содержит в себе информацию о:

- демографических и психографических особенностях пользователей (род занятий, пол, возраст, привычки, навыки работы с компьютером, мотивы и потребности);
- задачах или целях пользователя;
- контекстах (среде), в которой пользователь работает (рабочее место, операционная система, браузер, даже освещение на рабочем месте может учитываться).

Вооружившись «персонажем», нужно проектировать не для абстрактного посетителя, а для живого человека, с его пожеланиями и ограничениями.

Портрет персонажа. Портрет – это общее описание персонажа безотносительно информационной системы: фотография, имя, возраст, пол, профессия, личностные характеристики, которые могут повлиять на взаимодействие с сайтом, взгляды и интересы в интересующей нас сфере.

Цели персонажа. При взаимодействии персонажа с информационной системой он преследует какие-то цели. Выделяют следующие типы:

1) Цели, не связанные с информационной системой – это цели, которые персонаж может достичь без информационной системы.

2) Цели, связанные с информационной системой:

а) Личные — это цели, которые человек полностью разделяет и имеет мотивацию их достичь.

б) Навязанные – это цели, которые человеку диктует нечто извне, например, его компания, люди в его окружении или законодательство и которые он может не разделять.

Взаимодействие с продуктом. Здесь необходимо рассказать, как персонаж будет взаимодействовать с информационной системой:

- в какой обстановке будет использовать;
- как часто будет использовать информационную систему;
- был ли опыт использования подобных информационных систем;
- каковы факторы оценки информационной системы;
- каковы ожидания от поведения и содержания информационной системы;
- каковы ожидаемые результаты от взаимодействия.

Проверка и определение типов персонажей.

После описания персонажей необходимо проверить их на полноту/избыточность описания. Возможно было что-то упущено – например, важная цель или фактор оценки – или напротив рассмотрено что-то лишнее. Необходимо описывать поменьше деталей, не имеющих прямого отношения к информационной системе и взаимодействию с ней.

После создания расширенного списка персонажей выделяют два типа:

Ключевой персонаж – именно для него будет происходить проектирование информационной системы.

Второстепенный персонаж – его потребности необходимо учитывать, но их реализация не должна мешать ключевым персонажам реализовывать свои.

Ключевой персонаж должен объединять основные характеристики приоритетной целевой группы, совмещать в себе максимум её целей.

8. Разработка карты сценариев.

Сценарий поведения – это ситуация взаимодействия персонажа с продуктом. Сценарий описывает действия персонажа, цели, которые он хочет достичь, и ожидания от производимых действий.

Сценарии помогают нам смоделировать максимально короткий и эффективный способ достижения персонажем своих целей на сайте и, что также важно, предлагать пользователю контекстную информацию на его пути следования по сценарию. Основная польза сценариев в том, что они помогают избегать проектирования под исключительные ситуации.

Сценарий основан на цели/потребности пользователя и поэтому не позволяет от неё отклоняться при проектировании.

При составлении сценария необходимо учитывать следующее:

1) Эффективность сценария определяется его охватом, а не глубиной. Глубина определяется, когда моделируется поведение персонажа.

2) Сценарий нужно описать полностью: от начала до логического завершения.

3) Сценарий должен быть максимально правдоподобным.

Виды сценариев:

1) *Контекстные* – описывают контекст использования информационной системы пользователем, как правило, это описание на значительном промежутке времени – например, в течение одного рабочего дня. Такие сценарии обычно создаются для программного обеспечения или для сайта, который служит рабочим инструментом.

2) *Рабочие* – сценарии, которые детализируют использование продукта и описывают конкретные операции. Рабочие сценарии в свою очередь делятся на три типа:

- а) *типичные* – наиболее часто повторяющиеся. такие сценарии должны быть в дизайне на самом видном месте. этих сценариев редко бывает больше трёх;
- б) *исключительные* – реализуются исключительно редко. для таких сценариев взаимодействие можно проектировать грубо, без особого внимания к деталям;
- с) *обязательные* – без них невозможно решить некоторые задачи сайта.

Дополнительная функция сценариев, помимо описания ситуаций, — это уточнение и проверка целей персонажа. Сформулировать все цели персонажа максимально чётко при его описании довольно трудно.

Особое внимание необходимо обращать на контекстные (если они есть) и типичные сценарии. Они составляют основу взаимодействия пользователя и системы.

Обязательные, но не типичные сценарии, также важны, но они, как правило, реализуются достаточно редко.

Второй этап. Юзабилити-тестирование.

Юзабилити-тестирование может быть реализовано экспертами (экспертный подход) или пользователями (пользовательский подход).

Экспертный подход. При экспертном подходе в качестве пользователей выступают два и более экспертов (оптимальное количество для больших проектов 5-6 человек). Эксперты проходят сценарии поведения и анализируют их с точки зрения:

- стандартов юзабилити для конкретного типа программного продукта (например, Android Material Design для мобильных приложений на платформе Android);
- общих принципов юзабилити (эвристики Якоба Нильсона);
- здравого смысла и житейского опыта.

Преимущества экспертного подхода:

- быстрый в применении;
- эксперты гарантировано понимают общие задачи программного продукта.

Недостаток данного подхода – субъективизм (эксперты не являются реальными пользователями).

Пользовательский подход. Более качественный подход юзабилити-тестирования – пользовательский. Пользователям (5-10 человек из каждого сегмента целевой аудитории, оптимально – 7), согласившимся участвовать в тестировании, предлагают пройти наиболее распространенные и наиболее проблемные сценарии. Эксперт протоколирует действия пользователя, в идеальном варианте – фиксирует все в формате видео, чтобы отследить реакцию (эмоции) пользователя, но эксперт никак не влияет на действия пользователя.

Преимущества пользовательского подхода:

- объективные результаты (участвуют реальные пользователи);
- процесс легко измерим.

Возможные измерения при юзабилити-тестировании:

- время выполнения задачи;
- успешность выполнения задачи;
- эффект первого впечатления (например, сколько раз улыбнулся).

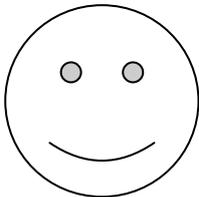
Недостатки пользовательского подхода:

- длительный по времени;
- дорогой (если пользователей привлекают на платной основе);
- большое внимание следует уделить подбору пользователей, в связи с чем предварительно составляются персонажи, формулируются требования по подбору пользователей из разных сегментов целевой аудитории.

Порядок выполнения работы

1. Выбрать программный продукт для исследования его целевой аудитории. Составить общую характеристику целевой аудитории.
2. Сегментировать целевую аудиторию в зависимости от параметров, наиболее влияющих на проектируемую систему (пол, возраст, род занятий, уровень компьютерной грамотности и т.п.).
3. В каждом сегменте выделить типичную персону.
4. Разработать карту эмпатии для этой персоны на основании рисунка 2.1.
5. Разработать профиль «ключевого персонажа» и «второстепенного персонажа» согласно таблице 2.1:

Таблица 2.1 – Профиль персонажа

<i><Категория целевой аудитории></i>	
<i><Роль персоны></i>	
	Описание <i><В описании указываются следующие характеристики: ФИО, пол, возраст, род занятий, семейное положение, образование, увлечения, социальный статус, место работы></i>
Фотография	
Личные характеристики	
Цели: <i><Перечень целей, которые пользователь стремится достичь во время использования системы></i>	
Взаимодействие с продуктом <i><Рабочий процесс и контекст (окружение)></i>	
Неудовлетворенности и ожидания <i><описание исключительных ситуаций и вытекающих из них проблем, описание дополнительных возможностей программного продукта></i>	

6. Разработать сценарий взаимодействия персонажа с программным продуктом.
7. Составить карту сценариев на основании таблицы 2.2.

Таблица 2.2 – Карта сценариев

<i>Шаг</i>	<i><название шага 1></i>
Вопросы	<i><Перечислить все вопросы, которые могут возникнуть у персонажа при работе с продуктом на данном шаге></i>
Пожелания	<i><Перечислить пожелания, которые могут возникнуть у персонажа при работе с продуктом на данном шаге ></i>
Эмоции	<i><Какие эмоции возникнут у персонажа от взаимодействия с продуктом></i>
Комментарии	<i><Описание окружения, исходных данных></i>

8. Оформить отчет и защитить лабораторную работу.

Содержание отчета

1. Цель работы.
2. Краткие теоретические сведения.
3. Описание целевой аудитории и ее сегментов.
4. Карта эмпатии для типичного представителя целевой аудитории.
5. Профиль «ключевого персонажа» и «второстепенного персонажа».
6. Сценарий взаимодействия персонажа с системой.
7. Карта сценариев.
8. Выводы по работе.

Контрольные вопросы

1. Что такое юзабилити?
2. Какие ключевые аспекты (понятия) юзабилити?
3. Что такое целевая аудитория?
4. Для чего необходимо изучать целевую аудиторию?
5. Какие существуют способы сбора информации о целевой аудитории?
6. Какие данные о сегментах целевой аудитории необходимо указывать после проведения исследования?
7. Что такое эмпатия? Что такое карта эмпатии?
8. Из каких характеристик состоит профиль персонажа?
9. Какой персонаж называют ключевым, а какой второстепенным?
10. Что такое сценарий поведения?
11. Какие существуют виды сценариев?
12. Какие существуют подходы к юзабилити-тестированию?
13. Охарактеризуйте экспертный подход юзабилити-тестирования.
14. Охарактеризуйте пользовательский подход юзабилити-тестирования.

Лабораторная работа № 3. Анализ качества интерфейсов с помощью юзабилити-метрик

Цель работы: закрепление теоретических знаний и приобретение практических навыков по юзабилити-тестированию и оценке качества интерфейса на основе измерения показателей результативности, эффективности и удовлетворенности.

Теоретические сведения

Юзабилити-тестирование – это метод исследования качества продукта, который представляет собой эксперимент, в ходе которого участники, максимально приближенные к реальным пользователям, выполняют типичные для пользователей задачи путем взаимодействия с интерфейсом системы.

Смысл тестирования состоит в проверке качества интерфейса продукта независимо от того, тестируется готовое программное обеспечение, минимально функциональный макет или же бумажный прототип. Главная цель тестирования заключается в том, чтобы сделать продукт более простым и удобным для пользователей.

В процессе юзабилити-тестирования выявляются сильные и слабые места в интерфейсе для дальнейшего улучшения его в ходе итерационного процесса разработки, а также осуществляется оценка общего качества интерфейса (например, для выбора одного из двух возможных вариантов).

Для оценки качества интерфейса или пригодности использования продукции могут быть использованы следующие юзабилити-метрики:

- результативность (effectiveness);
- эффективность (efficiency);
- удовлетворенность (satisfaction).

1. Результативность (effectiveness)

Результативность (effectiveness) – это «степень реализации запланированной деятельности и достижения запланированных результатов».

Измерение результативности зависит от цели или подцели, а также от точности и полноты выполнения задания, которых может

достигнуть пользователь. Это говорит о том, что оценить результативность продукта можно с помощью измерения таких характеристик, как процент достигнутых целей или успешно выполненных заданий.

Общую результативность исследуемого продукта можно рассчитать по следующей формуле:

$$\bar{E} = \frac{\sum_{j=1}^R \sum_{i=1}^N n_{ij}}{R \cdot N} \cdot 100\%, \quad (1)$$

где:

\bar{E} – общая результативность продукта,

R – общее число респондентов,

N – число сценариев (целей),

n_{ij} – результат выполнения i -го сценария j -м пользователем; при этом $n_{ij} = 1$, если i -й сценарий j -м пользователем завершен успешно, и $n_{ij} = 0$, если i -й сценарий не завершен, и j -й пользователь не смог достичь цели.

По полученным данным можно сделать приблизительные выводы об общей результативности продукта:

- 0–50% – ужасный;
- 50–75% – плохой;
- 75–90% – средний;
- 90–100% – хороший.

2. Эффективность (efficiency)

Эффективность (efficiency) – это «ресурсы, затрачиваемые пользователем в целях обеспечения точной и полной реализации целей».

Эффективность оценивают по уровню результативности и величине израсходованных ресурсов. Ресурсы могут включать умственные или физические усилия, время, материалы или финансовые затраты. Что касается программных продуктов и информационных систем, ключом к изменению эффективности является время, затраченное на выполнение задания, которое характеризует скорость работы с продуктом.

Временную относительную эффективность исследуемого продукта можно рассчитать по следующей формуле:

$$\bar{P} = \frac{\sum_{j=1}^R \sum_{i=1}^N n_{ij} \cdot t_{ij}}{\sum_{j=1}^R \sum_{i=1}^N t_{ij}} \cdot 100\%, \quad (2)$$

где:

\bar{P} – общая эффективность продукта,

R – общее число респондентов,

N – число сценариев (целей),

n_{ij} – результат выполнения i -го сценария j -м пользователем; при этом $n_{ij} = 1$, если i -й сценарий j -м пользователем завершен успешно, и $n_{ij} = 0$, если i -й сценарий не завершен, и j -й пользователь не смог достичь цели;

t_{ij} – время, затраченное на выполнение i -го сценария j -м пользователем.

3. Удовлетворенность (satisfaction)

Удовлетворённость (satisfaction) – это положительное отношение пользователя к использованию продукта и отсутствие дискомфорта при работе с ним.

В отличие от эффективности и результативности, где оцениваются действия пользователя, удовлетворение ориентировано на субъективное мнение пользователя.

Для измерения характеристики удовлетворенности пользователя продуктом можно использовать два метода:

- формализованные анкеты или опросники с наличием рейтинговой шкалы;
- набор положительных и отрицательных слов (ассоциаций).

Метод №1. Формализованные анкеты или опросники.

Каждая анкета или опросник содержит ряд заявлений, отражающих субъективное мнение пользователя. Например, «Мне нравится, как выглядит интерфейс» или «Я считаю, что продукт слишком сложный».

Вес или числовой коэффициент соответствия зависит от ответа.

Вес по 5-бальной рейтинговой шкале можно оценивать по-разному: от –2 до +2, от 1 до 5 или от 0 до 4.

Вопросы, касающиеся продукта, могут быть как положительными, так и отрицательными. Ответы на вопросы могут иметь как положительный вес (от 0 до 4), так и отрицательный (от 4 до 0).

Таблица 3.1 – Вес ответа на положительный вопрос (от 0 до 4)

Нет	Скорее нет	Нейтрально	Скорее да	Да
0	1	2	3	4

Таблица 3.2 – Вес ответа на отрицательный вопрос (от 4 до 0)

Нет	Скорее нет	Нейтрально	Скорее да	Да
4	3	2	1	0

Среднюю удовлетворенность для рейтинговой шкалы с 5-бальной рейтинговой шкалой от 0 до 4 можно по оценить по следующей формуле:

$$\bar{S} = \frac{\sum_{j=1}^R \sum_{i=1}^{Q_+} \frac{p_{ij}^+}{4} + \sum_{j=1}^R \sum_{i=1}^{Q_-} \frac{p_{ij}^-}{4}}{(Q_+ + Q_-) \cdot R} \cdot 100\%, \quad (3)$$

где:

\bar{S} – средняя удовлетворенность,

R – общее число респондентов,

Q_+ – число положительных вопросов в анкете,

Q_- – число отрицательных вопросов в анкете,

p_{ij}^+ – вес ответа на положительный вопрос для i -го вопроса и j -го ответчика,

p_{ij}^- – вес ответа на отрицательный вопрос для i -го вопроса и j -го ответчика.

Метод №2. Набор положительных и отрицательных слов.

Данный метод позволяет раскрыть субъективные ассоциации, порожденные продуктом. Суть данного метода заключается в том, что пользователям предлагается набор слов, отображающих как положительный, так и отрицательный ассоциативный ряд. Например, «удобный – неудобный, быстрый – медленный» и т.п. При этом

отрицательные слова смешивают с положительными. Каждый пользователь из предложенного набора слов выделяет только те слова, которые отражают его взаимодействие с продуктом.

Общие результаты позволяют построить:

- диаграмму, отражающую общее ассоциативное впечатление пользователей от взаимодействия с продуктом;
- рассчитать общий процент удовлетворения от продукта по следующей формуле:

$$\bar{S}_A = \frac{\sum_{j=1}^R A_j^+}{\sum_{j=1}^R (A_j^+ + A_j^-)} \cdot 100\%, \quad (4)$$

где:

\bar{S}_A – процент удовлетворенности от продукта,

R – общее число респондентов,

A_j^+ – количество положительных ассоциаций от j -го респондента,

A_j^- – количество отрицательных ассоциаций от j -го респондента.

Порядок выполнения работы

1. Выбрать три программных продукта для исследования удобств их интерфейсов.
2. Продумать и записать не менее трех заданий (сценариев) для оценки результативности.
3. Продумать и записать не менее трех заданий (сценариев) для оценки эффективности. Для каждого задания также подобрать разумное время, необходимое для его выполнения.
4. Продумать и записать несколько положительных и отрицательных вопросов для оценки удовлетворенности.
5. Разработать бланки заданий самостоятельно или воспользоваться *Приложениями*.
6. Выбрав группу респондентов (не менее трех), рассчитать все метрики. Сделать выводы.

Контрольные вопросы

1. Что такое юзабилити-тестирование?
 2. Какие характеристики программного продукта исследуются в процессе юзабилити-тестирования?
 3. Что такое результативность, и как она рассчитывается?
 4. Что такое эффективность, и как она рассчитывается?
 5. Что такое удовлетворенность, и как она рассчитывается?
 6. Опишите метод формализованных анкет или опросников.
 7. Опишите метод положительных и отрицательных слов.
- Как построить диаграмму, отражающую общее ассоциативное впечатление пользователей от взаимодействия с продуктом.

Приложение 2

Оценка результативности (effectiveness)

Дата: _____

Группа: _____

Количество респондентов: _____

Задание № 1: поиск автомобильного раздела на портале mail.ru

Задание № 2: поиск автомобиля Пежо 408 с пробегом до 10 тыс. км

Респондент	Задание № 1	Задание № 2
	Результат выполнения (+/-)	Результат выполнения (+/-)
1.		
2.		
3.		
4.		
5.		
6.		
7.		
8.		
9.		
10.		
Итого:		

Общая относительная результативность:

Расчет по формуле (1)

Приложение 3

Оценка эффективности (efficiency)

Дата: _____

Группа: _____

Количество респондентов: _____

Задание № 1: поиск автомобильного раздела на портале mail.ru

Задание № 2: поиск автомобиля Пежо 308 с пробегом до 10 тыс. км

Респондент	Задание № 1		Задание № 2	
	Результат выполнения (+/-)	Время выполнения, сек.	Результат выполнения (+/-)	Время выполнения, сек.
1.				
2.				
3.				
4.				
5.				
6.				
7.				
8.				
9.				
10.				

Временная относительная эффективность продукта

Расчет по формуле (2)

Приложение 4

Оценка удовлетворенности (satisfaction)

Метод: «Ассоциации»

Пожалуйста, отметьте слова в соответствии со своими впечатлениями от работы с сайтом

Дата: _____

Группа: _____

Имя: _____ (Имя,

Фамилия) _____

сайт: _____

беспольный – быстрый – запутанный – интересный –
информативный – красивый – легкий – любительский –
медленный – неинтересный – неинформативный –
некрасивый – непонятный – непривлекательный –
неприятный – неудобный – неэффективный –
перегруженный – полезный – понятный – привлекательный
– приятный – простой – профессиональный – прямой –
радостный – скучный – сложный – современный –
удобный – устаревший – эффективный.

простой												
профессиональный												
прямой												
радостный												
скучный												
сложный												
современный												
удобный												
устаревший												
эффективный												
	респонденты											

Приложение 6

Оценка удовлетворенности (satisfaction)

Метод: «Ассоциации»

Пожалуйста, обобщите данные, полученные от всех респондентов.

Дата: _____

Группа: _____

Количество респондентов: _____

Сайт: _____

Позитивные ассоциации	Количество баллов	Негативные ассоциации	Количество баллов
быстрый		бесполезный	
интересный		запутанный	
информативный		любительский	
красивый		медленный	
легкий		неинтересный	
полезный		неинформативный	
понятный		некрасивый	
привлекательный		непонятный	
приятный		непривлекательный	
простой		неприятный	
профессиональный		неудобный	
прямой		неэффективный	
радостный		перегруженный	
современный		скучный	
удобный		сложный	
эффективный		устаревший	

Общий процент удовлетворенности

Расчет по формуле (4)

Приложение 7

Дата: _____

Группа: _____

Профили участников тестирования

Респондент	Пол	Возраст	Опыт использования Интернета	Частота использования интернета	Выполнялись ли покупки в Интернет ранее?	Выполнялся ли веб-поиск автомобилей ранее?
1.						
2.						
3.						
4.						
5.						
6.						
7.						
8.						
9.						
10.						

Классификация опыта использования Интернета:

- *новичок* (никогда не пользовался или пользовался 1 или максимум 2 раза);
- *средний* (пользуется нерегулярно, меньше 3 лет);
- *большой* (пользуется ежедневно, в течение, по крайней мере, двух часов в день, умеет без проблем выполнять поиск);
- *очень большой* (разрабатывал веб-сайты, владеет HTML).

Приложение 8

Результаты тестирования (общее, индивидуальное)

Группа: _____

Имя _____ (Фамилия,
Имя): _____

Наиболее важные положительные аспекты веб-сайта:

1. Аспект 1. Описание
2. Аспект 2. Описание
3. Аспект 3. Описание

.....

Самые важные моменты усовершенствования веб-сайта:

1. Момент 1. Описание
2. Момент 2. Описание
3. Момент 3. Описание

.....

Основываясь на выявленных проблемах, рекомендуется:

1. Рекомендация 1. Описание
2. Рекомендация 2. Описание
3. Рекомендация 3. Описание

.....

Бельский Вадим Алексеевич

**ИНФОРМАЦИОННАЯ АРХИТЕКТУРА
И ЮЗАБИЛИТИ**

**Практикум
по выполнению лабораторных работ
по одноименному курсу для слушателей
специальности переподготовки
1-40 01 74 «Web-дизайн и компьютерная графика»
заочной формы обучения**

Подписано к размещению в электронную библиотеку
ГГТУ им. П. О. Сухого в качестве электронного
учебно-методического документа 28.02.20.

Пер. № 104Е.
<http://www.gstu.by>