

Министерство образования Республики Беларусь
Учреждение образования
«Белорусский государственный университет
информатики и радиоэлектроники»
Факультет компьютерного проектирования
Кафедра инженерной психологии и эргономики

ЭРГОНОМИКА ИНФОРМАЦИОННЫХ СИСТЕМ

*Рекомендовано УМО по образованию в области информатики
и радиоэлектроники в качестве пособия для специальности
1-58 01 01 «Инженерно-психологическое обеспечение
информационных технологий», направления специальности
1-40 05 01-09 «Информационные системы и технологии (в обеспечении
промышленной безопасности)»*

Минск БГУИР 2018

УДК 004:331.101.1(076.5)

ББК 32.973.202я73

Э74

Авторы:

Т. В. Калилец, В. С. Осипович, И. Ф. Киринович,
К. Д. Яшин, Е. А. Бурков, Н. А. Назаренко, П. И. Падерно

Рецензенты:

кафедра информационных систем управления
Белорусского государственного университета
(протокол №4 от 26.10.2017);

заведующий лабораторией компьютерной графики государственного научного учреждения «Объединенный институт проблем информатики Национальной академии наук Беларуси», кандидат технических наук, доцент В. В. Ткаченко

Эргономика информационных систем : пособие / Т. В. Калилец
Э74 [и др.]. – Минск : БГУИР, 2018. – 74 с. : ил.
ISBN 978-985-543-400-0.

Пособие содержит материалы к лабораторным занятиям, включающим вопросы и задания. Изложенные материалы способствуют приобретению студентами навыков проведения исследований юзабилити и эргономической экспертизы информационных систем различной сложности и назначения.

УДК 004:331.101.1(076.5)
ББК 32.973.202я73

ISBN 978-985-543-400-0

© УО «Белорусский государственный университет информатики и радиоэлектроники», 2018

СОДЕРЖАНИЕ

Введение.....	4
Лабораторная работа №1	
Сравнительный анализ информационных систем	6
Лабораторная работа №2	
Анализ пользователей. Проектирование ролей и полномочий	13
Лабораторная работа №3	
Построение UML-диаграмм	20
Лабораторная работа №4	
Проектирование графического интерфейса пользователя	33
Лабораторная работа №5	
Разработка прототипа пользовательского интерфейса	39
Лабораторная работа №6	
Выбор типа элементов управления для заданного набора полей	43
Лабораторная работа №7	
Исследование эргономичности информационной системы	52
Лабораторная работа №8	
Исследование информационной системы с использованием систем видеоокулографии.....	64
Список использованных источников.....	72

ВВЕДЕНИЕ

Одной из важнейших задач при разработке информационной системы является проектирование взаимодействия человека с ней. Специалисты-системотехники владеют навыками проектирования пользовательских интерфейсов, умеют проводить исследование юзабилити и эргономическую экспертизу информационных систем различной сложности и назначения. Разработчик учитывает, что интерфейс пользователя информационной системы должен удовлетворять ряду критериев: минимальное время выполнения задачи пользователем; минимальное число произвольных ошибок пользователя; минимальная неоднозначность в понимании интерфейса (способствует самообучению пользователей и делает их поведение предсказуемым); высокая стандартизация интерфейса (облегчает обучение пользователей); объем вводимой пользователем информации должен стремиться к минимуму; простота и визуальная привлекательность. Поэтому проектирование конкретного вида деятельности пользователя предусматривает: исследование внутренних средств деятельности человека (его опыта, знаний, навыков, восприятия, мышления, памяти); согласование этих средств с внешними средствами (документами, алгоритмами, инструментами, органами ручного управления) в соответствии с основной целью функционирования создаваемой информационной системы; формирование на основе разработанного проекта требований к техническим средствам системы, которые используются человеком; реализацию требований при создании продукта.

Данное пособие поможет освоить основные принципы проектирования пользовательского интерфейса информационной системы.

Особенностью пособия является изучение замкнутого цикла проектирования пользовательского интерфейса информационной системы. Кроме того, перечень лабораторных работ дополнен практическими заданиями, направленными на развитие аналитических навыков и навыков проведения эргономических исследований и экспертиз у студентов. Специалисты-системотехники владеют навыками анализа технического задания, проектирования информационной архитектуры, использования инструментов прототипирования пользовательского интерфейса, разработки UML-диаграмм, проведения юзабилити-исследований и эргономической экспертизы для информационных систем различного назначения. Это позволит реализовать ряд задач, необходимых специалистам в профессиональной деятельности: инженерно-психологическая оценка и проектирование программных и аппаратных средств; проведение системного анализа и определение инженерно-психологических требований к инфор-

мационным системам; работа с технической литературой, технологической документацией, научно-техническими отчетами, справочными материалами и другими информационными источниками.

Настоящее пособие разработано в соответствии с программой курса «Эргономика информационных систем» и призвано помочь студентам освоить принципы проектирования взаимодействия пользователя и информационной системы, а также более глубоко изучить теоретические и практические вопросы, связанные с обеспечением эргономичности информационных систем.

Библиотека БГУИР

Лабораторная работа №1

СРАВНИТЕЛЬНЫЙ АНАЛИЗ ИНФОРМАЦИОННЫХ СИСТЕМ

Цель работы: формирование у студентов знаний и навыков проведения сравнительного анализа информационных систем.

Теоретические сведения

Определим информационную систему (ИС) через ее основные функции (рисунок 1.1): ввод информации; обработка информации; вывод информации.



Рисунок 1.1 – Основные функции информационной системы

Минимальная единица информации, хранимая и обрабатываемая информационной системой, называется записью. Многие операции, выполняемые информационными системами в процессе обработки информации, используют несколько записей одновременно.

Сама запись может иметь внутреннюю структуру. Составляющие (элементы) записи обычно называются полями. Информационная система при обработке записи работает со всеми полями записи.

Три функции присутствуют в любой ИС, хотя могут иметь рудиментарные формы (например, в предметном указателе книги сбор информации и ее обработка были выполнены единственный раз, а вывод осуществляется перелистыванием книги читателем). Почти всякая отдельная программа может рассматриваться как информационная система. Например, текстовый процессор позволяет ввести информацию, он ее обрабатывает. Долговременным хранением информации для текстового процессора занимается операционная система. В текстовом процессоре возможен поиск информации. Текстовый процессор умеет выводить информацию.

Информационные системы всегда связаны с какой-то деятельностью человека (организации): расчетом траектории ракеты; управлением движением самолетов; дозировкой лекарств, вводимых больному; расчетом заработной платы; учетом недвижимости; поиском веб-страниц; реконструкцией археологических объектов и др.

Деятельность, связанная непосредственно с информационными системами, редко бывает основной (если только организация не занята исключительно разработкой и/или сопровождением ИС). Информационная система всегда обслуживает только основную деятельность организации/человека.

Наличие тесной связи информационной системы и обслуживаемой ею деятельности позволяет говорить о предметной области ИС – объектах той деятельности, с которой эта ИС связана, и отношениях между этими объектами. Так, в библиотечной ИС объектами предметной области являются издания (книги, журналы, эстампы, музыкальные записи и др.), средства хранения изданий (хранилища и стеллажи), читатели, библиографы и др. А в кадрово-бухгалтерской информационной системе объектами предметной области будут сотрудники, должности, рабочее время, штатное расписание, премии и надбавки, налоги и пр.

Каждая функция информационной системы может выполняться отдельным компонентом ИС. Такой компонент называется подсистемой или модулем (в зависимости от произвольно оцениваемой сложности или размера компонента). В небольших ИС подсистема может реализовать несколько функций; в больших и сложных ИС их функции детализируются (простейший пример – разделение функций хранения и обработки информации). Каждая такая детальная функция может реализовываться своей подсистемой; подсистемы могут реализовывать несколько различных детальных функций (относящихся, например, к одному из видов информации, обрабатываемой ИС). Например, подсистема расчета заработной платы в бухгалтерской ИС может реализовывать все три функции ИС, но по отношению только к некоторой части финансовой информации (используемой при расчете заработной платы, но не требующейся, например, для учета движения оборудования).

Для того чтобы подсистемы ИС могли реализовывать функции ИС, необходимы компоненты, согласованно используемые всеми или, по крайней мере, несколькими подсистемами. Такие компоненты называются обеспечениями или видами обеспечения. Различают следующие виды обеспечений.

1 Аппаратное – это компьютеры, сканеры, принтеры, синтезаторы звука, цифровые микрофоны, кассовые аппараты, устройства отображения информации, устройства управления датчиками физических величин и считывания дан-

ных с них, кабели и оборудование телекоммуникационных сетей, аппаратура электропитания и вентиляции и др.

2 Программное – операционные системы, языки программирования, системы управления базами данных, информационно-поисковые системы, библиотеки программных компонентов, серверное программное обеспечение. В программное обеспечение информационных систем никогда не включаются средства их разработки (редакторы программных текстов, компиляторы и др.).

3 Лингвистическое – словари данных, метаинформация, искусственные языки, языки форматных преобразований, описания коммуникативных форматов и др.

4 Информационное – полупостоянная информация, нормативно-справочная информация. Информационное и лингвистическое обеспечение иногда объединяют, включая лингвистическое обеспечение в информационное или наоборот.

5 Организационное – производственные роли, руководства пользователей и администраторов ИС.

Для реализации каждой функции информационной системы могут использоваться все или только часть обеспечений.

Информационные системы не существуют вечно – они создаются, работают (эксплуатируются) и замещаются другими информационными системами. Период от появления замысла информационной системы до ее полного замещения другой ИС называется жизненным циклом информационной системы. Структуры жизненных циклов различных ИС бывают разными, а чаще всего они либо линейны, когда одна стадия жизненного цикла последовательно сменяет другую, либо представляют собой спираль, когда стадии жизненного цикла сменяют друг друга, неоднократно повторяя некоторую последовательность стадий.

Линейный жизненный цикл информационной системы состоит из трех стадий: разработка (создание, производство); эксплуатация и сопровождение (использование и доработка); замещение другой информационной системой (с сохранением накопленных данных).

Линейный жизненный цикл в настоящее время характерен для военных и других информационных систем, связанных с использованием определенного оборудования (например, мобильных телефонов; с выработкой ресурса оборудования ИС замещается вместе с оборудованием) или высокими требованиями к качеству ИС (управление воздушным движением, обеспечение жизнедеятельности пациента в больнице и др.). Существенным элементом линейного жизненного цикла информационной системы является так называемое сопровождение системы. Процесс сопровождения включает две разновидности мероприятий.

1 Администрирование – мероприятия, направленные на поддержание приемлемых эксплуатационных характеристик ИС (используемые ресурсы, надежность и др.).

2 Сопровождение разработки – мероприятия, имеющие целью изменение характеристик ИС (прежде всего обнаружение и исправление ошибок; но также и модификация ИС для решения новых задач, не предусмотренных при ее разработке, или для обеспечения возможности эксплуатации ИС в условиях, которые также не были предусмотрены, например, на иной аппаратуре).

Большинству современных информационных систем присущ спиральный жизненный цикл. В спиральном жизненном цикле информационной системы эксплуатация ИС может быть не связана с процессом сопровождения разработки (однако от администрирования все равно никуда не деться). Ошибки, обнаруженные в процессе эксплуатации, и требования изменений, которые необходимо внести в информационную систему, фиксируются в фазе оценки информационной системы и поступают к разработчикам, которые через определенные интервалы времени выпускают новый вариант информационной системы, называемый версией (редакцией, релизом и т. п.). С получением очередной версии ИС эксплуатационный персонал замещает ею предыдущую версию системы. В реальности фазы эксплуатации, оценки и разработки могут совмещаться во времени.

Использование информационных систем со спиральным жизненным циклом позволяет: сократить время от начала разработки до начала эксплуатации ИС (за счет ограничения функциональности первой версии ИС); относительно быстро (с задержкой, равной времени выпуска очередной версии, которое может быть равным, например, даже двум неделям) реагировать на обнаруживаемые ошибки, изменяющиеся требования пользователей и изменяющиеся условия эксплуатации информационной системы.

Выполнение лабораторной работы

1 Изучить теоретическую часть.

2 Выбрать для анализа десять информационных систем, выполняющих одинаковые функции и имеющих сходные задачи. Разработать краткие характеристики для каждой информационной системы. Заполнить таблицу 1.1.

3 Разработать десять критериев для взаимного сравнения выбранных информационных систем. Подготовить обоснование выбора критериев и объяснение каждого из десяти выбранных критериев. Заполнить таблицу 1.2.

4 Разработать шкалу оценок критериев, сделать ее подробное описание. Заполнить таблицу 1.3.

- 5 Провести сравнительный анализ информационных систем.
- 6 Результаты сравнительного анализа оформить в виде таблицы 1.4.
- 7 Провести анализ полученных данных.
- 8 Разработать рекомендации по улучшению пяти информационных систем, которые набрали наименьшее количество баллов. Заполнить таблицу 1.5.
- 9 Оформить отчет.
- 10 Защитить лабораторную работу.

Таблица 1.1 – Характеристика выбранных для анализа информационных систем

Название информационной системы	Краткая характеристика информационной системы
1	
2	
3	
4	
5	
6	
7	
8	
9	
10	

Таблица 1.2 – Критерии оценки информационных систем

Название критерия	Описание критерия
1	
2	
3	
4	
5	
6	
7	
8	
9	
10	

Таблица 1.3 – Шкала оценки критериев

Обозначение	Описание
<i>Максимальная оценка</i>	
<i>Средняя оценка</i>	
<i>Минимальная оценка</i>	

Таблица 1.4 – Сравнительный анализ информационных систем

ИС	К 1	К 2	К 3	К 4	К 5	К 6	К 7	К 8	К 9	К 10
ИС 1										
ИС 2										
ИС 3										
ИС 4										
ИС 5										
ИС 6										
ИС 7										
ИС 8										
ИС 9										
ИС 10										
Примечание: ИС – информационная система; К – критерий										

Таблица 1.5 – Рекомендации по улучшению информационных систем

Название информационной системы	Рекомендации по улучшению
1	
2	
3	
4	
5	

Содержание отчета

- 1 Титульный лист.
- 2 Цель работы.
- 3 Краткие теоретические сведения.

- 4 Характеристика выбранных для анализа информационных систем (см. таблицу 1.1).
- 5 Критерии оценки информационных систем (см. таблицу 1.2).
- 6 Разработанная шкала оценки критериев с подробным описанием (см. таблицу 1.3).
- 7 Результаты сравнительного анализа информационных систем по критериям (см. таблицу 1.4).
- 8 Анализ полученных данных в результате проведения сравнения.
- 9 Разработанные рекомендации по улучшению информационных систем (см. таблицу 1.5).
- 10 Выводы по работе.

Контрольные вопросы

- 1 Что такое информационная система?
- 2 Какими функциями обладают информационные системы?
- 3 Что такое предметная область информационной системы?
- 4 Что такое подсистема информационной системы?
- 5 Какими компонентами (обеспечениями) должна обладать информационная система? Приведите примеры.
- 6 Что такое жизненный цикл информационной системы?
- 7 Опишите линейный жизненный цикл информационной системы.
- 8 Опишите спиральный жизненный цикл информационной системы.

Лабораторная работа №2

АНАЛИЗ ПОЛЬЗОВАТЕЛЕЙ. ПРОЕКТИРОВАНИЕ РОЛЕЙ И ПОЛНОМОЧИЙ

Цель работы: формирование у студентов знаний и навыков проведения сегментации целевой аудитории информационной системы в зависимости от параметров, наиболее влияющих на проектируемую систему; формирование навыков заполнения профиля гипотетической персоны для каждого сегмента целевой аудитории; формирование навыков проектирования ролей пользователей и выделения их полномочий.

Теоретические сведения

Целевая аудитория представляет собой группу пользователей, на которую рассчитано содержание информационной системы. Они точно знают, в получении какой информации они заинтересованы и какой именно товар или услугу желают приобрести. Исследование пользователей позволяет оценить рамки аудитории интернет-проекта, как реальные, так и потенциальные, с учетом возможного роста и изменения.

Знание целевой аудитории полезно в решении следующих задач.

1 Планирование стратегии развития интернет-маркетинга. Исследование аудитории позволяет понять, какие запросы интересны пользователям.

2 Размещение рекламы. Знание целевой аудитории позволяет размещать рекламу на тех ресурсах, которые интересны потенциальным клиентам, и оптимально использовать рекламный бюджет.

3 Работа с социальными медиа. Знание интересов аудитории позволяет подобрать площадки для общения с пользователями и решить поставленные задачи интернет-маркетинга.

Выявление целевой аудитории также поможет скорректировать видение проекта и выкинуть все лишнее, сэкономив средства на разработку, а также понять, что привлекает пользователей у конкурентов. Исследование целевой аудитории затрагивает текущих пользователей для их удержания и потенциальных пользователей, если задача состоит в привлечении целевой аудитории.

Для получения данных, составляющих портрет целевого посетителя, используют различные способы сбора информации. Существуют следующие способы сбора информации об аудитории.

1 Лог-анализатор сервера и данные счетчика посещений – позволяют изучить все действия пользователей и конкретизировать распределение аудитории по регионам, по времени и др.

2 Опросы аудитории – анкетирование пользователей, использование опросной формы или регистрации.

3 Опросы аудитории на сайтах опросов, совмещение панельных данных и данных счетчика посещений (анкетирование происходит на сайте панели).

4 Системы аудита и традиционные опросы исследовательских компаний.

Важным источником информации о целевой аудитории является 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). На основе данных, полученных в результате сбора информации, можно получить сводные цифры и изучить закономерности поведения групп пользователей и оценить эффективность рекламного воздействия.

Еще одним методом исследования целевой аудитории являются «персонажи», или «персоны». «Персона» – это обобщенный профиль представителя целевой аудитории, содержащий 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 и т. п.).

Итоговая цель исследования – получение полного портрета целевой аудитории. В ходе исследования собирают и анализируют полученные данные (из социальных сетей и других публичных источников), проводятся социологические исследования, анализ посещаемости и многое другое.

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

– структура как сегментированная целевая аудитория – из каких групп состоит;

– объем – какое количество людей является целевой аудиторией;

– предпочтения – какие предпочтения свойственны целевой аудитории в целом и ее отдельным группам в частности;

– факторы принятия решения о покупке – какие факторы влияют на принятие решения о покупке товара и какие моменты являются определяющими;

– социально-демографические характеристики – каковы ключевые характеристики целевой аудитории: пол, возраст, социальный статус и т. п.;

– тенденции – как будет изменяться объем, состав и предпочтения целевой аудитории в ближайшем будущем.

Проектирование роли пользователей и полномочий выполняется следующим образом. В ролевой модели понятие *субъект* замещается понятиями *пользователь* и *роль*.

Роль – активно действующая в системе абстрактная сущность, с которой связан ограниченный, логически связанный набор полномочий, необходимых для осуществления определенной деятельности. При использовании ролевой политики управление доступом осуществляется в две стадии: для каждой роли указывается набор полномочий, представляющий набор прав доступа к объекту; каждому пользователю назначается список доступных ему ролей.

Полномочия назначаются ролям в соответствии с принципом наименьших привилегий. Из которого следует, что каждый пользователь должен обладать только минимально необходимым для выполнения своей работы набором полномочий.

Ролевая модель описывает систему в виде множеств (рисунок 2.1):

- U – множество пользователей;
- R – множество ролей;
- S – множество сеансов работы пользователя с системой;
- P – множество полномочий на доступ к объекту.

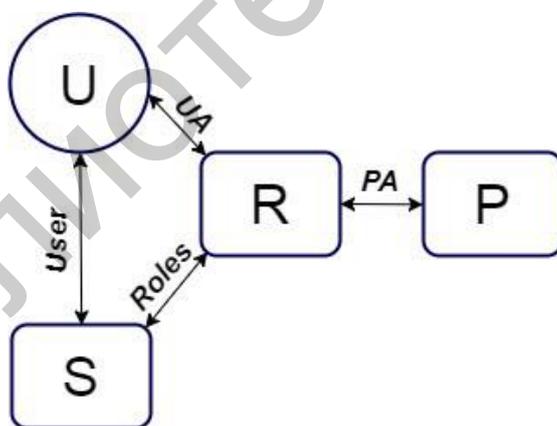


Рисунок 2.1 – Ролевая модель, используемая для описания системы

PA отображает множество полномочий на множество ролей, устанавливая для каждой роли набор присвоенных ей полномочий. UA отображает множество пользователей на множество ролей, определяется для каждого пользователя набор доступных ему ролей.

Правило управления доступом в ролевой политике безопасности определяется следующими функциями:

$$\text{User: } S \rightarrow U.$$

Для каждого сеанса S определяет пользователя U , который осуществляет этот сеанс работы с системой $\text{user}(S) = U$.

$$\text{Role: } S \rightarrow R.$$

Для каждого сеанса S эта функция определяет набор ролей R из множества, которые могут быть одновременно доступны пользователю в этом сеансе.

$$\text{Permissions: } S \rightarrow P.$$

Для каждого сеанса S задает набор доступных в нем полномочий, которые определяются как совокупность полномочий всех ролей, заданных в этом сеансе.

Критерий безопасности в ролевой модели. Система считается безопасной, если любой пользователь U системы, работающий в сеансе S , может осуществлять действия, требующие полномочия P , только в том случае, если

$$P = \text{permissions}(S).$$

Следствие критерия: управление доступом может осуществляться главным образом не с помощью назначения полномочий, а путем задания отношений UA , назначающих роли пользователям и функции roles , определяющих доступным в сеансе набор ролей.

Иерархическая ролевая модель. Это наиболее распространенный тип ролевой модели, в которой роли упорядочиваются по уровню предоставленных полномочий. Иерархия ролей подразумевает, что если пользователю присвоена некоторая роль, то ему автоматически назначаются и все подчиненные ей по иерархии роли. Таким образом, каждому пользователю назначается некоторое подмножество иерархии ролей, а в каждом сеансе доступна совокупность полномочий ролей, составляющих фрагмент этой иерархии. Как результат имеем уменьшение размерности роли.

Группирование ролей и полномочий. Роли и полномочия, которые дополняют друг друга и назначение которых по отдельности не имеет смысла, объединяются в группы, которые могут быть назначены только целиком. Для этого вводятся дополнительные правила, в соответствии с которыми любая роль может быть назначена пользователю только в том случае, если ему уже присвоен определенный набор ролей, а роль может быть наделена полномочиями только тогда, когда с ней уже связан определенный набор полномочий.

Ролевая политика управления доступом не гарантирует безопасности с помощью формального доказательств, а только определяет характер ограничений, соблюдение которых служит критерием безопасности.

Выполнение лабораторной работы

- 1 Изучить теоретическую часть.
- 2 Выбрать информационную систему для анализа пользователей, проектирования ролей и полномочий.
- 3 Выделить десять пользователей информационной системы, основываясь на ее функциональных возможностях. Разработать описание каждой роли пользователя, работающего с информационной системой. Заполнить таблицу 2.1.
- 4 Разработать перечень из десяти полномочий (разрешенных действий и функций) для всех ролей пользователей, работающих с информационной системой. Разработать описание разрешенных действий. Заполнить таблицу 2.2.
- 5 Разработать комментарии, отображающие доступность возможностей информационной системы в зависимости от роли каждого пользователя.
- 6 Оформить отчет.
- 7 Защитить лабораторную работу.

Таблица 2.1 – Роли пользователей системы

Пользователь	Описание роли пользователя
1	
2	
3	
4	
5	
6	
7	
8	
9	
10	

Таблица 2.2 – Полномочия пользователей

Роль, работающего с информационной системой	Описание полномочий пользователя, работающего с информационной системой
1	1 2 3 4 5 6 7 8 9 10
...	
...	
10	1 2 3 4 5 6 7 8 9 10

Содержание отчета

- 1 Титульный лист.
- 2 Цель работы.
- 3 Краткие теоретические сведения.
- 4 Краткое описание выбранной информационной системы.
- 5 Описание ролей пользователя, работающего с информационной системой (см. таблицу 2.1).
- 6 Описание полномочий для каждой выделенной роли пользователя (см. таблицу 2.2).
- 7 Выводы по работе.

Контрольные вопросы

- 1 Дайте определение понятию «целевая аудитория».
- 2 Какие задачи позволяет решить анализ целевой аудитории?
- 3 Какие способы сбора информации о целевой аудитории вам известны?
- 4 Опишите метод исследования целевой аудитории «Персонажи».
- 5 Опишите ролевую модель, используемую для проектирования ролей пользователя и полномочий.
- 6 Опишите иерархическую ролевую модель.

Библиотека БГУИР

Лабораторная работа №3 ПОСТРОЕНИЕ UML-ДИАГРАММ

Цель работы: формирование у студентов знаний и навыков работы с языком моделирования UML; проведение концептуального проектирования системы на UML; проектирование взаимодействия пользователя с информационной системой.

Теоретические сведения

Унифицированный язык моделирования (Unified Modeling Language, UML) – это универсальный язык визуального моделирования систем. Унифицированный процесс (Unified Process, UP) – это методология, указывающая на исполнителей, действия и артефакты, которые необходимо использовать, осуществить или создать для моделирования программной системы. UML моделирует наше окружение как системы взаимодействующих объектов. Объект – это цельный блок, состоящий из данных и функциональности.

В UML-модели есть два аспекта:

- статическая структура – описывает, какие типы объектов важны для моделирования системы и как они взаимосвязаны;
- динамическое поведение – описывает жизненные циклы этих объектов и то, как они взаимодействуют друг с другом для обеспечения требуемой функциональности системы.

В UML существует четыре общих механизма, последовательно применяемых ко всему языку моделирования: спецификации, дополнения, принятые деления, механизмы расширения.

Модели UML имеют по крайней мере два измерения: графическое, позволяющее визуализировать модель с помощью диаграмм и пиктограмм, и текстовое, состоящее из спецификаций различных элементов модели (рисунок 3.1). Спецификации – это текстовые описания семантики элемента, формирующие семантический задний план, который объединяет модель и наполняет ее смыслом. Разработку моделей с помощью UML начинают с графической модели, которая позволяет визуализировать систему, а затем по мере ее развития добавляют в задний план семантику. Однако модель можно считать полезной или полной, только если семантика модели присутствует в семантическом заднем плане.

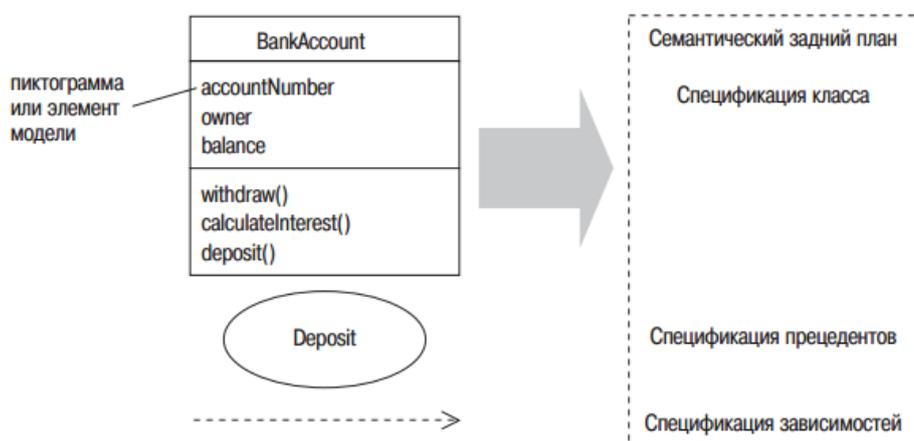


Рисунок 3.1 – Графическое и семантическое измерение UML-диаграмм

В UML каждый элемент модели обозначается простым символом, к которому можно добавлять ряд дополнений (рисунок 3.2), визуализирующих аспекты спецификации элемента. С помощью этого механизма видимая на диаграмме информация может быть представлена в соответствии с конкретными требованиями.

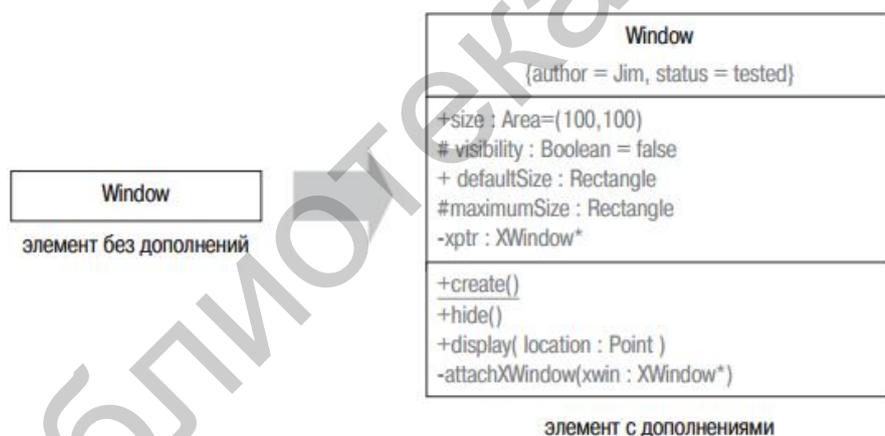


Рисунок 3.2 – Элементы модели UML-диаграмм

Принятые деления описывают конкретные способы представления мира. В UML существует два принятых деления: классификатор/экземпляр (классификатор – это абстрактное понятие; экземпляр – конкретная сущность) и интерфейс/реализация.

UML – расширяемый язык моделирования, который включает три простых механизма расширения:

- ограничения – расширяют семантику элемента, обеспечивая возможность добавления новых правил;
- стереотипы – обеспечивают возможность определения новых элементов модели UML на основании существующих; стереотипы добавляют новый элемент в метамодель UML;
- помеченные значения – предоставляют способ расширения спецификации элемента, обеспечивая возможность добавления в него новой специальной информации.

UML состоит из трех следующих структурных элементов.

1 Сущности – это элементы модели, среди которых выделяют структурные сущности – существительные UML-модели, такие как класс, интерфейс, кооперация, прецедент, активный класс, компонент, узел; поведенческие сущности – глаголы UML-модели, такие как взаимодействия, деятельности, автоматы; группирующая сущность – пакет, используемый для группировки семантически связанных элементов модели в образующие единое целое модули; аннотационная сущность – примечание, которое может быть добавлено к модели для записи специальной информации.

2 Отношения связывают сущности. Отношения определяют, как семантически связаны две или более сущностей.

3 Диаграммы – это представления моделей UML. Они показывают наборы сущностей, которые «рассказывают» о программной системе и являются нашим способом визуализации того, что будет делать система (аналитические диаграммы) или как она будет делать это (проектные диаграммы).

Существуют следующие типы UML-диаграмм (рисунок 3.3): диаграмма классов; диаграмма составных структур; диаграмма компонентов; диаграмма развертывания; диаграмма объектов; диаграмма пакетов; диаграмма деятельности; диаграмма взаимодействия; диаграмма последовательности; коммуникационная диаграмма; временная диаграмма; диаграмма прецедентов; диаграмма состояний.



Рисунок 3.3 – Типы UML-диаграмм

Рассмотрим диаграммы структуры.

1 **Диаграммы классов (class diagrams)** – это набор статических, декларативных элементов модели, предназначенных для моделирования структуры объектно-ориентированных приложений (рисунок 3.4). Класс (class) – категория вещей, которые имеют общие атрибуты и операции. Классы используются в процессе анализа предметной области для составления словаря предметной области разрабатываемой системы. Это могут быть как абстрактные понятия предметной области, так и классы, на которые опирается разработка и которые описывают программные или аппаратные сущности.

Диаграммы классов могут применяться и при прямом проектировании, то есть в процессе разработки новой системы, и при обратном проектировании – описании существующих и используемых систем. Информация с данной диаграммы напрямую отображается в исходный код приложения, таким образом, диаграмма классов представляет собой конечный результат проектирования и является отправной точкой процесса разработки.



Рисунок 3.4 – Диаграмма классов

2 Диаграммы компонентов (component diagrams) (рисунок 3.5) используются при моделировании компонентной структуры распределенных приложений; внутри каждый компонент может быть реализован с помощью множества классов. Они позволяют определить архитектуру разрабатываемой системы, установив зависимости между программными компонентами, в роли которых может выступать исходный и исполняемый код.

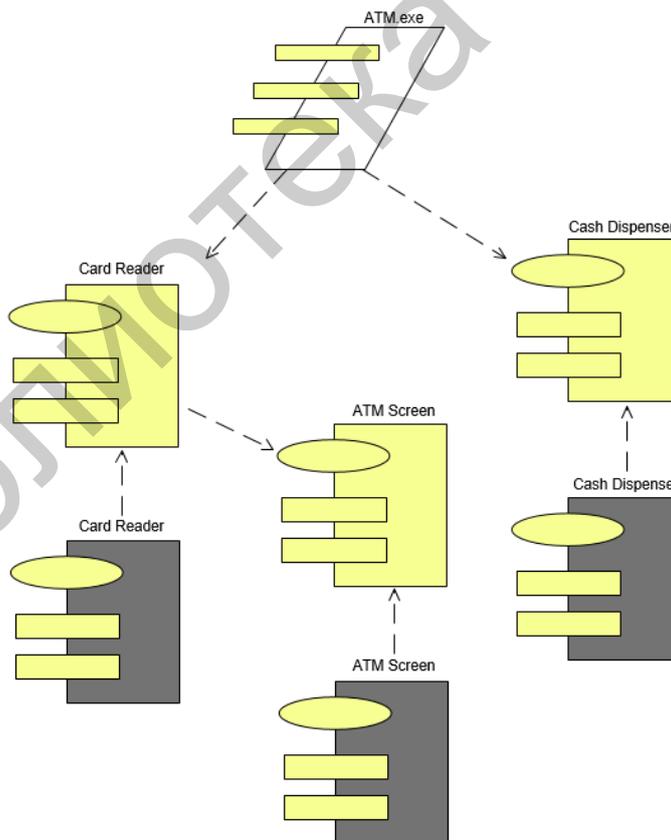


Рисунок 3.5 – Диаграмма компонентов

Основными графическими элементами диаграммы компонентов являются компоненты, интерфейсы и зависимости между ними. Диаграмма компонентов разрабатывается для следующих целей: визуализации общей структуры исходного кода программной системы; спецификации исполняемого варианта программной системы; обеспечения многократного использования отдельных фрагментов программного кода; представления концептуальной и физической схем баз данных.

В разработке диаграмм компонентов участвуют как системные аналитики и архитекторы, так и программисты. Диаграмма компонентов обеспечивает согласованный переход от логического представления к конкретной реализации проекта в форме программного кода. Одни компоненты могут существовать только на этапе компиляции программного кода, другие – на этапе его исполнения. Диаграмма компонентов отражает общие зависимости между компонентами, рассматривая последние в качестве классификаторов.

3 Диаграммы объектов (object diagrams) (рисунок 3.6) применяются для моделирования фрагментов работающей системы, показывают множество объектов-экземпляров классов (изображенных на диаграмме классов) и отношений между ними в некоторый момент времени. Являются основой для сценариев, описываемых диаграммами взаимодействия.

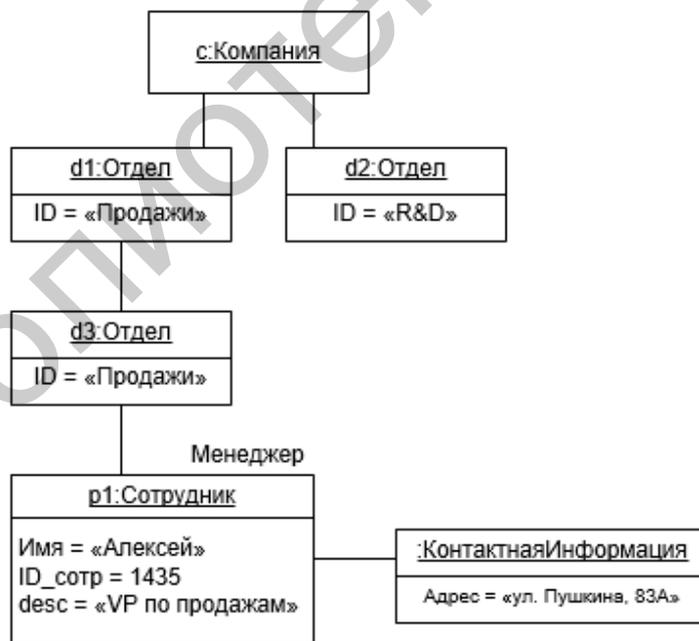


Рисунок 3.6 – Диаграмма объектов

4 Диаграммы композитных структур (composite structure diagrams) используются для моделирования составных структурных элементов моделей – коопераций, композитных компонентов и т. д.

5 Диаграммы развертывания (deployment diagrams) предназначены для моделирования аппаратной части системы, с которой программное обеспечение непосредственно связано (размещено или взаимодействует). При этом представляются только компоненты – экземпляры программы, являющиеся исполняемыми файлами или динамическими библиотеками. Те компоненты, которые не используются на этапе исполнения, на диаграмме развертывания не показываются.

Диаграмма развертывания (рисунок 3.7) содержит графические изображения процессоров, устройств, процессов и связей между ними. В отличие от диаграмм логического представления, диаграмма развертывания является единой для системы в целом, поскольку должна полностью отражать особенности ее реализации. Разработка диаграммы развертывания, как правило, является последним этапом спецификации модели программной системы. При разработке диаграммы развертывания преследуют следующие цели: определить распределение компонентов системы по ее физическим узлам; показать физические связи между всеми узлами реализации системы на этапе ее исполнения; выявить узкие места системы и реконфигурировать ее топологию для достижения требуемой производительности. Диаграммы развертывания разрабатываются совместно системными аналитиками, сетевыми инженерами и системотехниками.



Рисунок 3.7 – Диаграмма развертывания

6 Диаграммы пакетов (package diagrams) служат для разбиения объемных моделей на составные части, а также (традиционно) для группировки классов моделируемого программного обеспечения, когда их слишком много (рисунок 3.8).

Диаграммы пакетов могут использовать пакеты, содержащие прецеденты для иллюстрации функциональности программного обеспечения системы, а также пакеты, которые представляют различные слои программного комплекса для иллюстрации его слоистой архитектуры. Зависимости между этими пакетами могут быть снабжены метками/стереотипами, чтобы указать механизм связи между слоями.

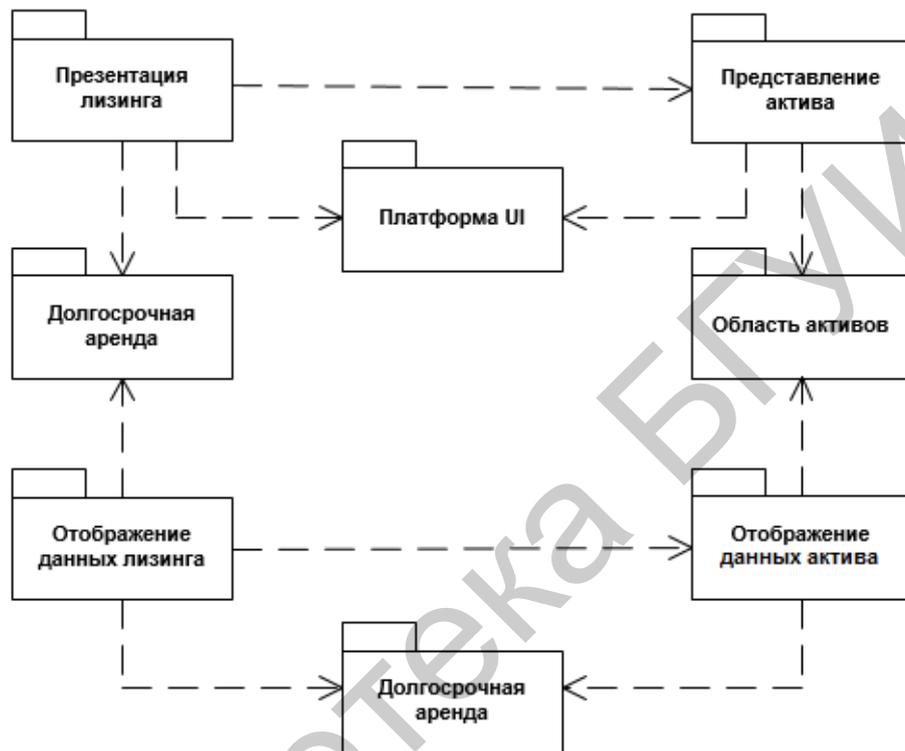


Рисунок 3.8 – Диаграмма пакетов

Рассмотрим поведенческие диаграммы.

1 Диаграммы активности (activity diagrams) (рисунок 3.9), называемые также диаграммами деятельности или диаграммами видов деятельности, представляют собой блок-схему, которая показывает, как поток управления переходит от одной деятельности к другой, при этом внимание фиксируется на результате деятельности. Результат может привести к изменению состояния системы или возвращению некоторого значения. Диаграмма деятельности отличается от традиционной блок-схемы более высоким уровнем абстракции; возможностью представления с помощью диаграмм деятельности управления параллельными потоками наряду с последовательным управлением. Одно из основных направлений использования диаграмм деятельности – отображение внутрисистемной точки зрения на прецедент. Диаграммы деятельности применяют для описания

шагов, которые должна предпринять система после того, как инициирован прецедент.

Разработка диаграммы деятельности преследует следующие цели: детализировать особенности алгоритмической и логической реализации прецедентов; выделить последовательные и параллельные потоки управления; подготовить детальную документацию для взаимодействия разработчиков системы с ее заказчиками и проектировщиками.

Графически диаграмма деятельности представляется в форме графа деятельности, вершинами которого являются состояния действия или состояния деятельности, а дугами – переходы от одного состояния действия/деятельности к другому. Каждая диаграмма деятельности должна иметь единственное начальное и единственное конечное состояния (на практике иногда можно видеть несколько конечных состояний на одной диаграмме, но это одно и то же состояние, изображенное несколько раз для лучшей читабельности диаграммы). Саму диаграмму деятельности принято располагать таким образом, чтобы действия следовали сверху вниз. В этом случае начальное состояние будет изображаться в верхней части диаграммы, а конечное – в ее нижней части.

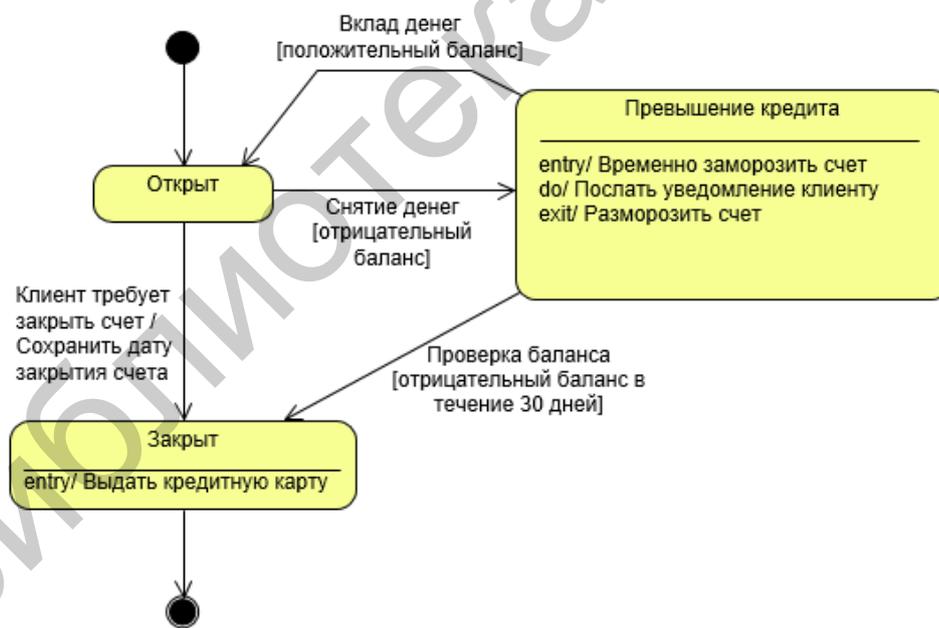


Рисунок 3.9 – Диаграммы активности

2 Диаграммы вариантов использования (use case diagrams) описывают функциональное назначение системы или то, что система должна делать.

Разработка диаграммы преследует следующие цели: определить общие границы и контекст моделируемой предметной области; сформулировать общие

требования к функциональному поведению проектируемой системы; разработать исходную концептуальную модель системы для ее последующей детализации в форме логических и физических моделей; подготовить исходную документацию для взаимодействия разработчиков системы с ее заказчиками и пользователями.

Суть диаграммы вариантов использования состоит в следующем. Проектируемая система представляется в виде множества сущностей или «актеров», взаимодействующих с системой с помощью вариантов использования. При этом «актером» (actor), или действующим лицом, называется любая сущность, взаимодействующая с системой извне. Это может быть человек, техническое устройство, программа или любая другая система, которая может служить источником воздействия на моделируемую систему так, как определит сам разработчик. Вариант использования служит для описания сервисов, которые система предоставляет «актеру». Диаграмма вариантов использования может дополняться пояснительным текстом, который раскрывает смысл или семантику составляющих ее компонентов (рисунок 3.10).

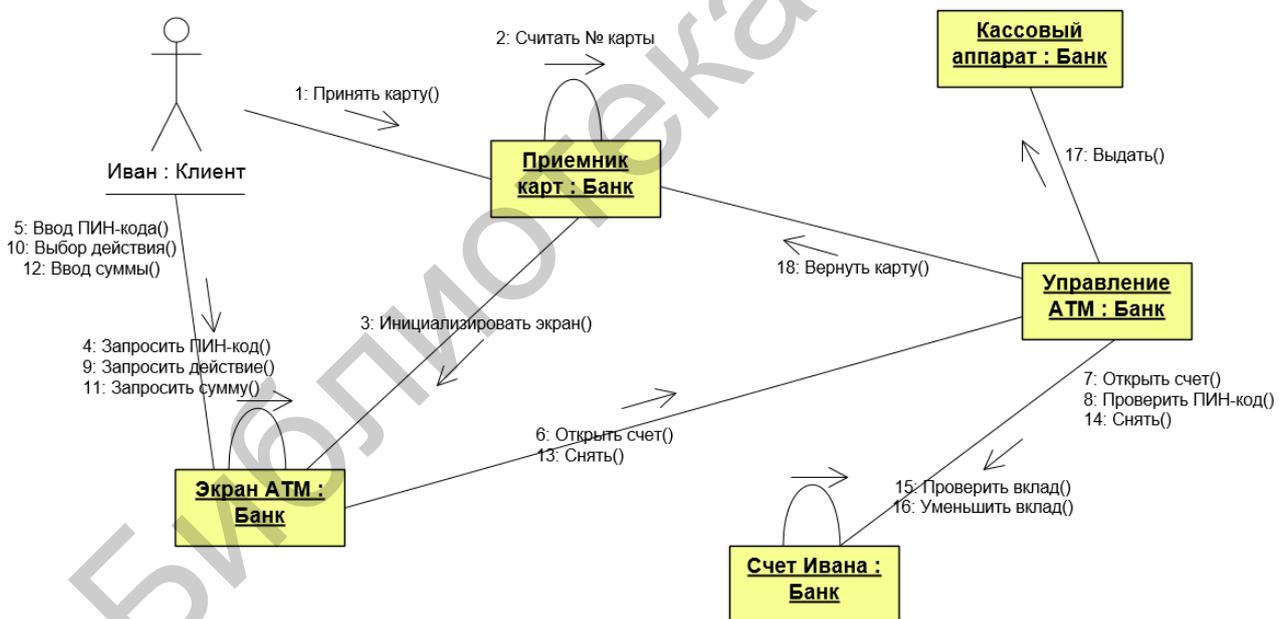


Рисунок 3.10 – Диаграмма вариантов использования

3 Диаграммы конечных автоматов (state machine diagrams) применяются для задания поведения реактивных систем.

4 Диаграммы взаимодействий (interaction diagrams):

- диаграммы последовательностей (sequence diagrams) используются для моделирования временных аспектов внутренних и внешних протоколов ПО;
- диаграммы схем взаимодействия (interaction overview diagrams) служат для организации иерархии диаграмм последовательностей;
- диаграммы коммуникаций (communication diagrams) являются аналогом диаграмм последовательностей, но по-другому изображаются (в привычной графовой манере);
- временные диаграммы (timing diagrams) являются разновидностью диаграмм последовательностей и позволяют в наглядной форме показывать внутреннюю динамику взаимодействия некоторого набора компонентов системы.

Выполнение лабораторной работы

- 1 Изучить теоретическую часть.
- 2 Выбрать две информационные системы. Сделать их описание, перечислив действующие лица и события (таблица 3.1).
- 3 Разработать диаграммы активности (с применением Visual Studio) для выбранных информационных систем. Выбрать начальное и конечное состояние деятельности. Разработать переходы от одного состояния к другому. Изобразить диаграмму активности в форме графа деятельности (см. рисунок 3.9).
- 4 Разработать диаграмму вариантов использования для выбранных систем (с применением Visual Studio) (см. рисунок 3.10). Описать системы, перечислив действующие лица и события. Разработать комментарии к диаграммам.
- 5 Разработать диаграммы классов для описания систем (см. рисунок 3.4). Разработать описание систем. Провести анализ систем, выделив сущности и отношения между ними. Изобразить диаграммы классов, применив Visual Studio, разработать комментарии к ним.
- 6 Разработать диаграммы объектов (см. рисунок 3.6). Выделить объекты-экземпляры и отношения между ними в некоторый момент времени, для каждой из выделенных информационных систем. Изобразить диаграмму объектов с помощью Visual Studio.
- 7 Разработать диаграммы развертывания (см. рисунок 3.7). Выделить аппаратную часть для каждой информационной системы, которая используется на этапе исполнения. Объяснить свой выбор. Разработать связи между компонентами. Изобразить диаграмму развертывания с помощью Visual Studio.
- 8 Оформить отчет.
- 9 Защитить лабораторную работу.

Таблица 3.1 – Характеристика информационных систем

Название информационной системы	Описание информационной системы	Действующие лица	События

Варианты заданий для выполнения лабораторной работы

Вариант 1. Цифровой диктофон | Общественный транспорт.

Вариант 2. Будильник | Университет.

Вариант 3. Телевизор | Компьютерная игра «Гонки».

Вариант 4. Система автоматизации для пункта проката | Система учета экспонатов в музее.

Вариант 5. Мини-АТС | Система учета пациентов в поликлинике.

Вариант 6. Телефон | Игра в шахматы.

Вариант 7. Стиральная машина | Школа.

Вариант 8. Таксофон | Завод.

Вариант 9. Домофон | Иерархия мебели.

Вариант 10. Холодильник | Система учета проживающих в общежитии.

Вариант 11. Кодовый замок | Компьютерная игра «Волейбол».

Вариант 12. Турникет метро | Музыкальный плеер.

Вариант 13. Система учета товаров | Аукцион.

Вариант 14. Библиотечная система | Супермаркет.

Вариант 15. Калькулятор | Интернет-форум.

Вариант 16. Пылесос | Система тестирования знаний.

Вариант 17. Система контроля состояния воздуха в помещении | Кондиционер.

Вариант 18. Обогреватель | Новостной сайт.

Вариант 19. Система выдачи реактивов | Кассовый аппарат.

Вариант 20. Принтер | Социальная сеть.

Содержание отчета

- 1 Титульный лист.
- 2 Цель работы.
- 3 Краткие теоретические сведения.
- 4 Описание выбранных информационных систем.
- 5 Диаграммы активности. Разработанные комментарии к ним.

- 6 Диаграммы вариантов использования. Разработанные комментарии к ним.
- 7 Диаграммы классов. Разработанные комментарии к ним.
- 8 Диаграммы объектов.
- 9 Диаграммы развертывания. Пояснения к выбранной аппаратной части.
- 10 Выводы по работе.

Контрольные вопросы

- 1 Перечислите диаграммы, относящиеся к структурным диаграммам.
- 2 Перечислите диаграммы, относящиеся к поведенческим диаграммам.
- 3 Охарактеризуйте диаграмму классов.
- 4 Что такое атрибут?
- 5 Какие виды связей могут устанавливаться на диаграммах между классами?
- 6 Охарактеризуйте диаграмму компонентов.
- 7 Охарактеризуйте диаграмму развертывания.
- 8 Охарактеризуйте диаграмму пакетов.
- 9 Охарактеризуйте диаграмму активности.
- 10 Охарактеризуйте диаграмму вариантов использования.
- 11 На какие виды подразделяются диаграммы взаимодействий?

Лабораторная работа №4

ПРОЕКТИРОВАНИЕ ГРАФИЧЕСКОГО ИНТЕРФЕЙСА ПОЛЬЗОВАТЕЛЯ

Цель работы: формирование у студентов знаний и навыков создания макета графического интерфейса пользователя; изучение ключевых принципов размещения элементов управления; ознакомление с основными элементами управления (виджетами).

Теоретические сведения

Графический интерфейс пользователя (GUI) – разновидность пользовательского интерфейса, в котором элементы интерфейса (меню, кнопки, значки, списки и т. п.), представленные пользователю на дисплее, исполнены в виде графических изображений. Графический интерфейс пользователя является частью пользовательского интерфейса и определяет взаимодействие с пользователем на уровне визуализированной информации.

Разработка интерфейса обычно начинается с определения задачи или набора задач, для которых продукт предназначен. Интерфейс должен быть ориентированным на пользователя, т. е. отвечать его нуждам и потребностям.

Создание приложения необходимо начинать с разработки макета интерфейса. Макет интерфейса – это визуальное статическое представление концепции интерфейса пользователя. Различают два вида макетов интерфейса.

1 Грубый макет. Макет низкой точности, полученный в результате проектирования интерфейсов. Грубые макеты призваны отражать порядок, структуру и расположение элементов на экране. Таким аспектам дизайна, как сетка, цвет, шрифты и т. д. при создании грубых макетов интерфейса внимание не уделяется. Грубые макеты интерфейса заказчик получает в виде динамичного прототипа приложения, который можно использовать для тестирования эргономичности или начала разработки приложения.

2 Законченный макет. Макет высокой степени четкости – результат разработки дизайна интерфейса с проработанным цветом, шрифтами, фоном, сеткой. Заказчик получит такой макет в виде графических файлов в формате Photoshop или в виде HTML- или XAML-кода.

Создание макета – это первый шаг создания интерфейса. На этом этапе приложение материализуется, а замыслы приобретают очертания готового продукта. Проектирование макета интерфейса необходимо для выполнения ряда задач:

- постановка задачи разработчикам;
- постановка задачи дизайнерам интерфейса; хорошо продуманный макет гарантирует, что конечный результат будет максимально близок к тому, чего хотел заказчик;
- обсуждение функционала приложения с заказчиком, удобнее обсуждать функционал, наглядно представленный с помощью грубых макетов; макет поможет заказчику получить четкое представление о том, что именно будет сделано;
- тестирование юзабилити; если макет достаточно проработан и интерактивен, т. е. на этом этапе чаще всего проводят тестирование с целью устранения проблем юзабилити-требований.

Разработку макета приложения начинают с составления структурной схемы информационной системы (рисунок 4.1). При разработке структуры информационной системы необходимо предусмотреть возможность дальнейшего ее расширения (добавления разделов и ссылок).

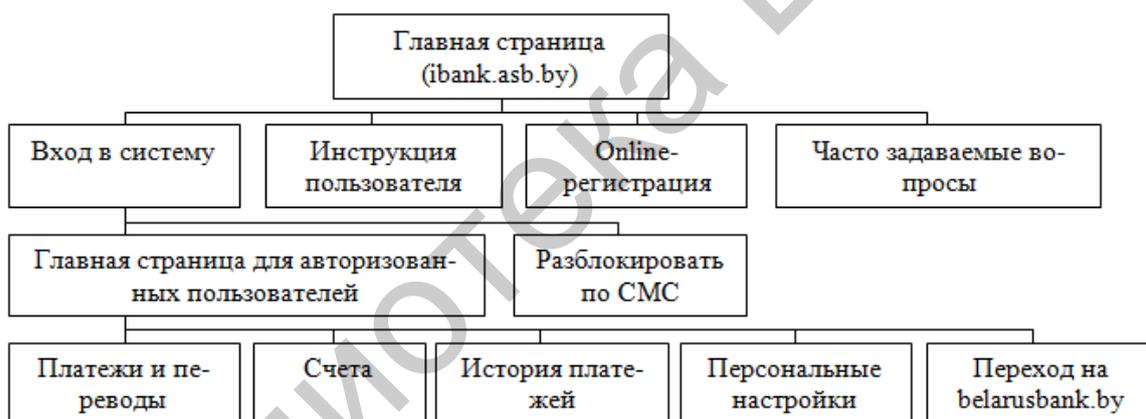


Рисунок 4.1 – Структурная схема информационной системы

Структурная диаграмма определяет также методы навигации, которые будут использоваться. Существует три типа навигации: перемещение по ссылкам; использование формы поиска; доступ через портал к его содержимому.

Требования к навигации информационной системы: должна быть логична по структуре и оформлению; должна иметь удобную основную и вспомогательную навигацию с одинаковым позиционированием на всех страницах; навигация должна в любой момент предоставлять посетителю возможность быстро и безошибочно определить свое местонахождение в информационной системе.

Перемещение по страницам и разделам информационной системы осуществляется посредством текстовых или графических ссылок. Текстовые ссылки легко редактировать (как редактирование текста самой ссылки, так и добавление новых), они совместимы со всеми браузерами, имеют высокую скорость загрузки. Графические ссылки наглядны и эффектны (ссылки привлекают внимание), однако если в браузере отключена загрузка графики, тогда посетитель страницы кнопку и ссылку не увидит.

Расположение ссылок на странице. Система навигации (меню) может располагаться слева (наиболее традиционное расположение), справа или сверху на странице. Если меню располагается слева или справа, а на странице много информации, тогда можно продублировать ссылки также внизу страницы, что даст пользователю дополнительную возможность перейти в другой раздел информационной системы, избежав дополнительных действий. Если в качестве ссылок используются кнопки, то желательно их продублировать вверху или внизу страницы на случай, если у пользователя отключен вывод графики. Нельзя убирать из меню пункт, ссылающийся на текущую страницу. Ссылки в меню должны быть такими, чтобы было сразу понятно, в какой раздел они ведут и какую информацию можно найти в этом разделе.

Карта сайта. Когда разработана структурная схема сайта, ее можно трансформировать в карту сайта, представив в виде многоуровневого списка, добавляя в нее ссылки к отдельным элементам и предоставляя непосредственный доступ к различным страницам информационной системы. Карта сайта используется для повышения юзабилити, обеспечения для пользователей дополнительного инструмента навигации. Если информационная система состоит более чем из двух-трех десятков страниц, полезно создать специальную страницу – карту сайта, содержащую ссылки на все остальные страницы. Карта сайта, как и главная страница, должна быть доступна со всех страниц.

В GUI пользователь имеет произвольный доступ (с помощью устройств ввода – клавиатуры, мыши, джойстика и т. п.) ко всем видимым экранным объектам (элементам интерфейса) и осуществляет непосредственное манипулирование ими. Элемент интерфейса – примитив графического интерфейса пользователя, имеющий стандартный внешний вид и выполняющий стандартные действия. Существует стандартный набор элементов интерфейса, включающий следующие элементы управления: кнопка (button); сдвоенная кнопка (split button); радиокнопка (radio button); флаговая кнопка (check box); значок (иконка, icon); список (list box); дерево – иерархический список (tree view); раскрывающийся список (combo box, drop-down list); метка (label); поле редактирования (text box, edit field); элемент для отображения табличных данных (grid view); меню (menu);

главное меню окна (main menu или menu bar); контекстное меню (popup menu); ниспадающее меню (pull down menu); окно (window); диалоговое окно (dialog box); модальное окно (modal window); панель (panel); вкладка (tab); панель инструментов (toolbar); полоса прокрутки (scrollbar); ползунок (slider); строка состояния (status bar); всплывающая подсказка (tooltip, hint). Встречаются и другие элементы управления, которые могут не входить в некоторые наборы: радиальное меню (pie menu или radial menu) – кольцевое меню вокруг указателя мыши (выбор пункта меню осуществляется движением указателя мыши в направлении пункта меню); кнопка последовательного выбора – элемент, значение в котором выбирается последовательным щелчком кнопки мыши по нему (в отличие от раскрывающегося списка, такая кнопка не позволяет видеть другие значения, кроме выбранного); счетчик – двунаправленный вариант для числовых значений (нажатие кнопки позволяет изменить значение параметра на единицу в большую или меньшую сторону); heads-up display – отображение поверх всех элементов значения каких-то параметров либо важных сообщений; пузырь – подсказка наподобие филактера в комиксах, которая указывает на элемент – источник сообщения; валкодер – вращающийся элемент управления наподобие ручки настройки во многих радиоприемниках (может быть как одно-, так и многооборотным); скрываемый виджет – элемент, позволяющий скрыть часть элементов управления, когда они не используются; индикатор уровня (level indicator) – элемент для индикации значения какой-либо величины (иногда вместо него используется индикатор процесса).

В дизайне элементов интерфейса требуется учитывать все: начиная от цвета, формы, пропорций, заканчивая когнитивной психологией.

При разработке макета интерфейса необходимо учитывать требования, заложенные в спецификации либо техническом задании на проектируемый продукт. Составляться данные документы должны с учетом требований следующих стандартов.

1 ISO/IEC/ IEEE 29148-2011 – обеспечивает единую трактовку процессов и продуктов, используемых при разработке требований на протяжении всего жизненного цикла систем и программного обеспечения.

2 ISO/IEC 26514:2008. Requirements for designers and developers of user documentation – требования для дизайнеров и разработчиков пользовательской документации.

3 ISO/IEC FDIS 18019:2004. Guidelines for the design and preparation of user documentation for application software – рекомендации по созданию документации пользователя.

4 IEEE Std 1063-2001. IEEE Standard for Software User Documentation – в документе обозначены требования к структуре, содержанию и формату инструкций пользователя.

5 IEEE STD 830-1998 – описывает содержание и качественные характеристики правильно составленной спецификации требований к программному обеспечению (SRS) и приводит несколько шаблонов SRS. Описанная методика имеет своей целью установление требований к разрабатываемому программному обеспечению, а также может применяться, чтобы помочь в выборе собственных и коммерческих программных изделий.

Выполнение лабораторной работы

- 1 Изучить теоретическую часть.
- 2 Разработать краткую характеристику выбранной информационной системы. Выделить основные функции информационной системы (таблица 4.1).
- 3 На основании выделенных функций разработать структурную схему информационной системы.
- 4 Разработать карту навигации для выбранной системы (многоуровневый список).
- 5 На карте навигации в зависимости от специфики системы выделить разделы, доступные различным пользователям в зависимости от роли. Описать условия перехода между разделами (таблица 4.2).
- 6 Используя графический редактор на выбор (Microsoft Visio 2010, Axure, Adobe Photoshop, Balsamiq, Casoo), разработать три макета графического интерфейса пользователя.
- 7 Для разработанных макетов подготовить текстовое описание элементов управления (таблица 4.3).
- 8 Оформить отчет.
- 9 Защитить лабораторную работу.

Таблица 4.1 – Характеристика информационной системы

Название информационной системы	Характеристика информационной системы	Основные функции информационной системы

Таблица 4.2 – Разделы информационной системы и описание переходов между ними

Роль информационной системы	Разделы, доступные для выделенной роли	Описание условий перехода между разделами

Таблица 4.3 – Описание элементов управления макета информационной системы

Название элемента управления	Тип	Условия видимости	Условия доступности	Описание
				<i>Формат, допустимые значения, максимальная и минимальная длина, поведение</i>

Содержание отчета

- 1 Титульный лист.
- 2 Цель работы.
- 3 Краткие теоретические сведения.
- 4 Краткая характеристика информационной системы.
- 5 Структурная схема информационной системы.
- 6 Карта навигации информационной системы.
- 7 Описание условий перехода между разделами.
- 8 Макеты графического интерфейса пользователя.
- 9 Описание элементов управления.
- 10 Выводы по лабораторной работе.

Контрольные вопросы

- 1 Дайте определение понятию макет интерфейса.
- 2 Какие разновидности макетов интерфейса существуют? Охарактеризуйте каждый вид.
- 3 Для чего необходимо проектирование макета интерфейса?
- 4 Какие типы навигации существуют?
- 5 Что необходимо учитывать в дизайне элементов интерфейса?
- 6 Для чего необходима карта сайта?

Лабораторная работа №5

РАЗРАБОТКА ПРОТОТИПА ПОЛЬЗОВАТЕЛЬСКОГО ИНТЕРФЕЙСА

Цель работы: формирование у студентов практических навыков и знаний прототипирования пользовательских интерфейсов, создания прототипа страницы веб-приложения; формирование знаний и навыков создания адаптивных интерфейсов; развитие техники выбора реализации интерфейсов для различных типов устройств.

Теоретические сведения

Адаптивный веб-дизайн (англ. Adaptive Web Design) – дизайн веб-страниц, обеспечивающий правильное отображение сайта на различных устройствах, подключенных к Интернету, и динамически подстраивающийся под заданные размеры окна браузера.

Целью адаптивного веб-дизайна является универсальное отображение содержимого веб-сайта для различных устройств. Для того чтобы веб-сайт был удобно просматриваемым с устройств разных форматов и экранов различных разрешений, по технологии адаптивного веб-дизайна не нужно создавать отдельные версии веб-сайта для отдельных видов устройств. Один сайт может работать на смартфоне, планшете, ноутбуке и телевизоре с выходом в Интернет, т. е. на всем спектре устройств (рисунок 5.1).



Рисунок 5.1 – Отображение адаптивного шаблона на различных устройствах

Адаптивный макет базируется на использовании медиазапросов для адаптации контента под различные параметры экранов. Наиболее популярный вариант создания блоков на основе Twitter Bootstrap для 960 или 1200 пикселей, где ширину всего будущего контента можно распределить по 9 или 12 колонкам.

Как правило, при разработке адаптивных интерфейсов на основе Bootstrap используют следующие размеры экранов по ширине в пикселях (рисунок 5.2):

- 480 и ниже для смартфонов;
- 768 и выше для планшетов;
- 980 и выше для стандартного экрана;
- 1200 и выше для широкоформатного экрана.

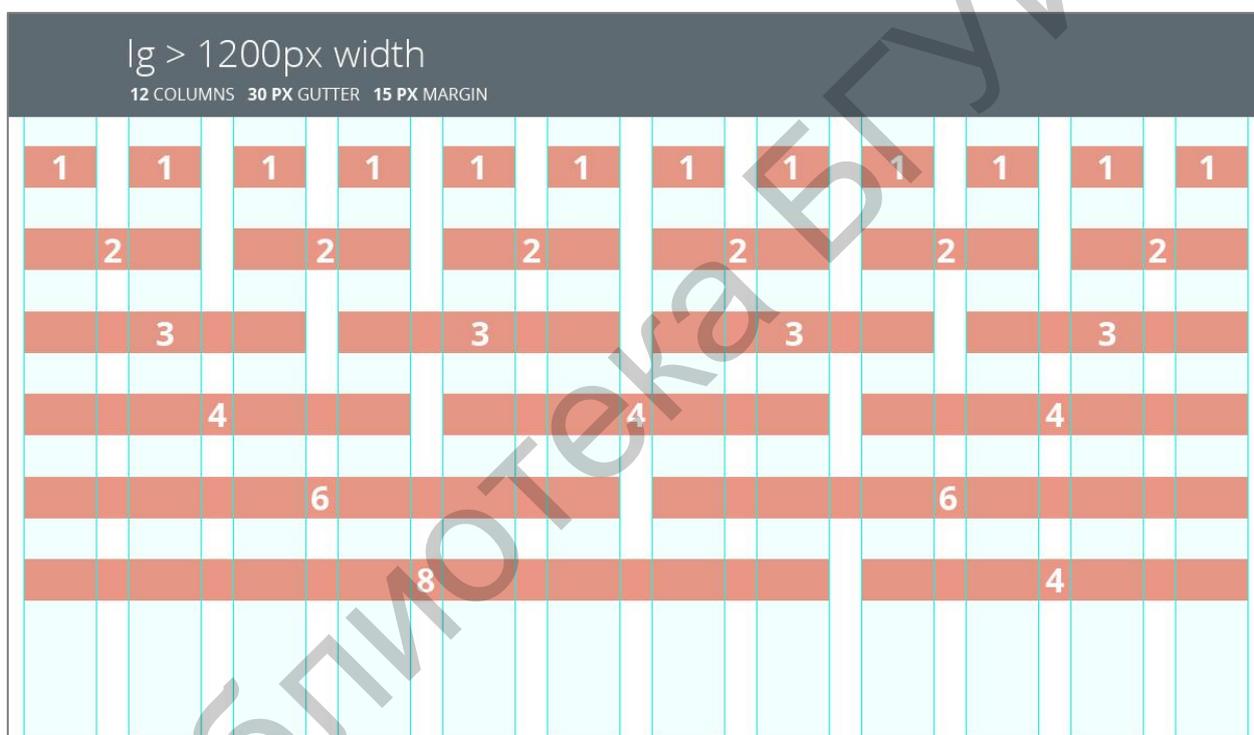


Рисунок 5.2 – Примеры комбинаций блоков Bootstrap

Хотя технология адаптивной верстки и решила задачу отображения контента на мобильных устройствах, она все же имеет определенные ограничения. Дальнейшим эволюционным развитием адаптивной верстки служит ее частная реализация на основе относительных, а не фиксированных единиц измерения контента. Это, в свою очередь, позволяет получить динамическое изменение сетки страницы, размеров элементов, а также изменение вариантов отображения самих элементов с целью предоставления лучшего опыта взаимодействия в каждом отдельном случае (рисунок 5.3).

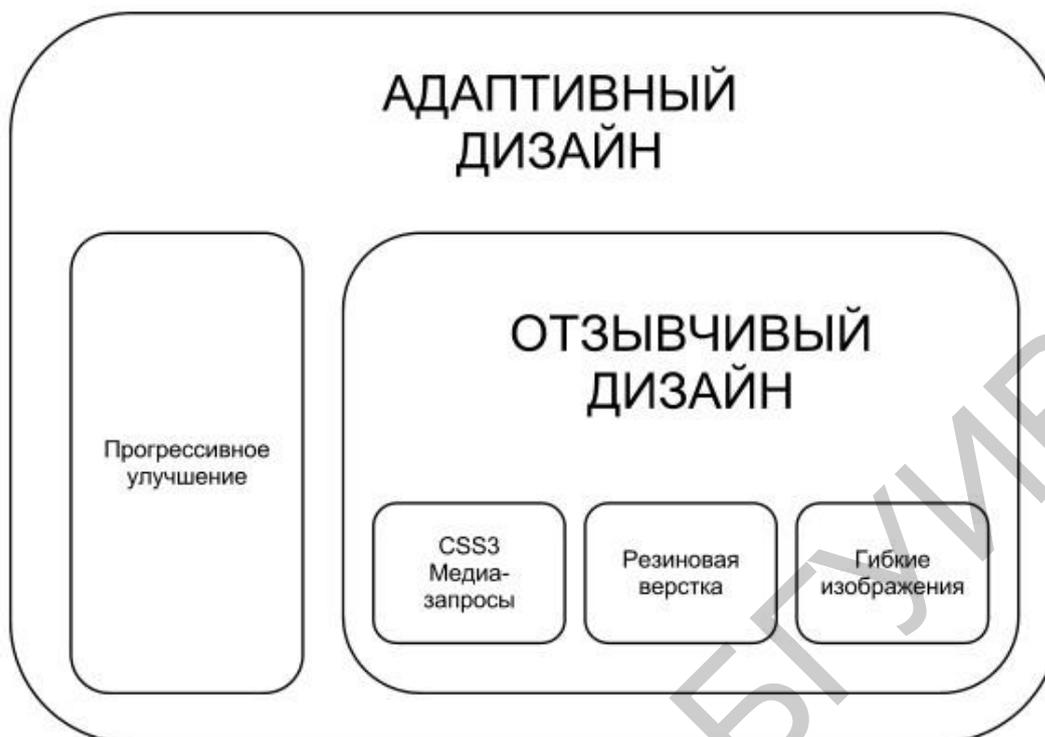


Рисунок 5.3 – Соотношение и состав понятия адаптивный дизайн

Инструменты, используемые для прототипирования, представлены в таблице 5.1.

Таблица 5.1 – Инструменты прототипирования

Название инструмента	Лицензия
Axure RP	Бесплатно для образовательных учреждений
Balsamiq Mockups	30 дней бесплатно
Draw.io	Бесплатно
Sketch	Только для Mac
MS Visio	Расширенный пакет Microsoft Office

Таким образом, использование отзывчивого дизайна позволяет более эффективно задействовать пространство рабочей области, т. к. убираются жесткие пороги, заранее заложенные в стандартной адаптивной верстке с фиксированным указанием ширины элементов, что необходимо при большом разнообразии разрешений устройств, получающих доступ к интерфейсу.

Выполнение лабораторной работы

- 1 Изучить теоретическую часть.
- 2 Описать выбранную информационную систему.
- 3 Выбрать один из инструментов прототипирования, перечисленных в таблице 5.1 (или использовать любой графический редактор с настраиваемыми границами).
- 4 Выбрать вариант исполнения интерфейса.
- 5 Определить три разрешения экранов, для которых будут создаваться шаблоны.
- 6 Создать Bootstrap-сетку для выбранных разрешений.
- 7 Создать основные информационные блоки. Расположить их на сетке для выбранных разрешений экранов.
- 8 Оформить отчет.
- 9 Защитить лабораторную работу.

Содержание отчета

- 1 Титульный лист.
- 2 Цель работы.
- 3 Краткие теоретические сведения.
- 4 Краткое описание информационной системы.
- 5 Прототипы с замещенными информационными блоками.
- 6 Выводы по работе.

Контрольные вопросы

- 1 Что такое адаптивный дизайн?
- 2 Какую проблему решает применение адаптивного дизайна при разработке интерфейсов?
- 3 Для чего используется Twitter Bootstrap?
- 4 Какой принцип распределения элементов на сетке Twitter Bootstrap?
- 5 В чем преимущество использования отзывчивого дизайна перед адаптивной версткой?

Лабораторная работа №6 ВЫБОР ТИПА ЭЛЕМЕНТОВ УПРАВЛЕНИЯ ДЛЯ ЗАДАННОГО НАБОРА ПОЛЕЙ

Цель работы: формирование у студентов знаний и практических навыков проектирования элементов управления для заданного набора полей; изучение стандартных элементов управления и приобретение практических навыков выбора типа элементов управления для заданного набора полей.

Теоретические сведения

Элемент управления – примитив графического интерфейса пользователя. Он имеет стандартный внешний вид (рисунок 6.1) и предназначен для ввода/вывода информации и управления работой программы. Элементы управления пользовательского интерфейса обеспечивают всю необходимую и достаточную функциональность, логичны, интуитивно понятны.

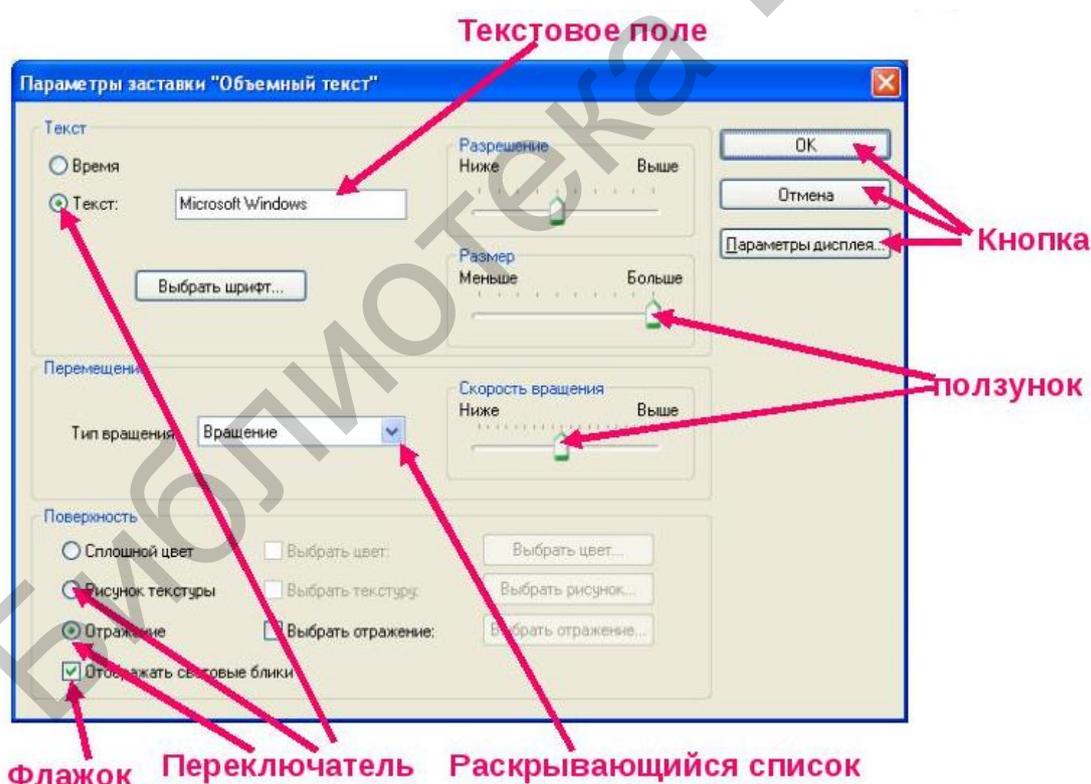


Рисунок 6.1 – Элементы управления

Программный продукт с правильно подобранными элементами управления обеспечивает повышение конкурентоспособности продукта; увеличение

аудитории продукта; уменьшение затрат на обучение и поддержку пользователей; уменьшение потерь продуктивности работников при внедрении системы и более быстрое восстановление утраченной продуктивности и др.

Управляющие элементы. В приложениях целесообразно располагать элементы таким образом, чтобы наиболее важные были заметны пользователю в первую очередь.

В большинстве языков текст располагается справа налево, сверху вниз. Принцип работы с информацией таким образом используется и при работе с компьютером: взгляд пользователя сперва упирается в верхний левый угол монитора, поэтому более важные элементы должны находиться там. Например, если информация в окне относится к покупателю, то поле с его именем должно быть расположено там, где его увидят в первую очередь. Кнопки ОК и CANCEL, напротив, должны располагаться в нижней части экрана: пользователь не нажимает их до тех пор, пока не закончит работать с остальными данными в окне.

Создавая приложение, разработчик должен стремиться к тому, чтобы пользователь увидел «позволительность» там, где это требуется. Например, трехмерные эффекты на кнопках подчеркивают их предназначение. Но если сделать ее плоской, то пользователь может не догадаться, что это кнопка. Если убрать элемент объемности из поля для ввода текста, его легко можно принять за редактируемый элемент.

Использование свободного пространства – это важный фактор. С помощью свободного пространства можно отделять друг от друга элементы в окне и придавать особый акцент некоторым из них. Не стоит располагать очень много элементов на одном окне – это ведет к ощущению хаотичности интерфейса и создает впечатление недоработанной, дешевой программы.

Наиболее важным принципом создания интерфейса является стремление к простоте. Если интерфейс выглядит сложным, то программа будет восприниматься пользователем с трудом. С эстетической точки зрения простой, ясный дизайн всегда выглядит лучше.

Стандартная ошибка при составлении интерфейса программ – это попытка смоделировать внешний вид программы по подобию реальных объектов. Предположим, что требуется сделать программу для ввода больших анкет (например, для страховой компании). Первое, что хочется сделать, – это разработать окно, которое в точности повторяет анкету. Но это вызовет массу проблем. Ведь размеры монитора и формат анкеты часто не совпадают. Следовательно, это скажется на интерфейсе программы: если поля для ввода будут слишком маленькими, пользователю придется постоянно работать с полосами

прокрутки. В любом случае это плохой интерфейс. В ситуациях такого рода надо разработать собственный интерфейс, сохранив вид анкеты для печати.

Элементы управления по форме в верстке делятся на точечные, линейные и прямоугольные.

Точечные элементы. Компактные элементы интерфейса с выраженным центром внимания – пиктограмма, асинхронный индикатор прогресса, чек-бокс.

Линейные элементы. Большая часть элементов интерфейса – линейные, хорошо работающие в строке: переключатель, кнопка, поле ввода, ползунок, звездочки рейтинга.

Прямоугольные элементы. К ним относят текстовые области, интерактивные диаграммы, цветовые палитры, карты – эти элементы управления имеют прямоугольную форму. Благодаря занимаемой площади такие элементы становятся самыми заметными якорными объектами.

Существует стандартный набор элементов управления, который может иметь различные названия и специфические особенности в разных средствах проектирования пользовательских интерфейсов, однако общее их назначение и выполняемые функции схожи. К этому набору можно отнести следующие группы элементов (рисунок 6.2): кнопки; списки; поля ввода/вывода; меню; окна.

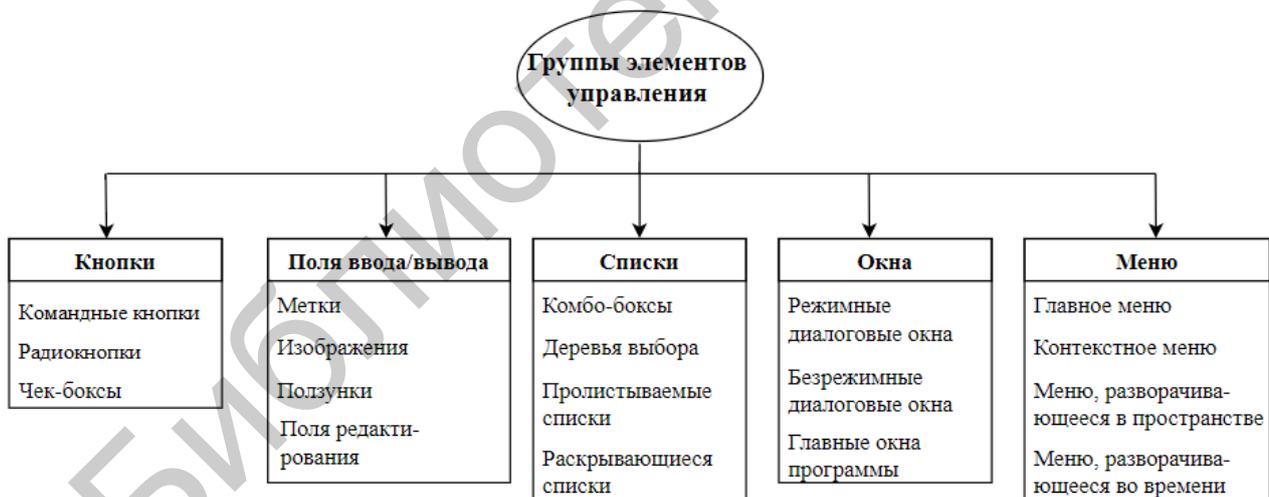


Рисунок 6.2 – Основные элементы графического интерфейса

В группе кнопок можно выделить такие элементы, как командная кнопка (button), радиокнопка (radiobutton), переключатель (check box). Командная кнопка отвечает за выполнение какого-либо действия по ее нажатии. Радиокнопка позволяет выбрать одну опцию (пункт) из predetermined набора (группы). Переключатель позволяет осуществлять как множественный выбор из

предопределенного набора, так и выбор из двух состояний (активно/неактивно) у одного элемента.

К группе списков относятся следующие элементы: раскрывающийся список (drop-down list), пролистываемый список (list box), дерево выбора (tree view), комбо-бокс (combo box). Раскрывающиеся списки показывают все свои элементы по нажатию (как правило, на первый элемент). Содержание пролистываемого списка видно сразу. Пролистываемые списки могут позволять пользователям совершать как единственный, так и множественный выбор (список единственного/множественного выбора). Дерево выбора представляет собой совокупность связанных отношениями структуры элементов в иерархическом дереве. Каждый элемент (узел) может иметь несколько подразделов. Комбо-бокс является гибридом списка с полем ввода: пользователь может выбрать существующий элемент либо ввести свой. Группа логически взаимосвязанных комбо-боксов образует расширенный комбо-бокс.

К полям ввода/вывода относятся: поле редактирования (text box), метка (label), изображение (picture box), ползунок (trackbar). Поле редактирования позволяет вводить информацию в программу, а также выводить ее и редактировать существующую. Поле редактирования может быть как однострочным, так и многострочным. Метку также можно использовать для вывода информации, но чаще этот компонент используется как предустановленная статическая надпись. Ползунок позволяет вводить информацию в предопределенном диапазоне значений.

Меню – метод взаимодействия пользователя с системой, при котором пользователь выбирает команду из предложенных вариантов, а не предоставляет системе свою. Существует главное меню (menu strip) и контекстное меню (context menu strip). Главное меню постоянно присутствует на экране и содержит, как правило, основные функции программы. Контекстное меню становится доступным после какого-либо действия (как правило, по нажатию второй кнопки мыши) и предоставляет функции для решения некоторых конкретных задач. Оба вида меню могут иметь иерархическую структуру (содержать подразделы).

Окна используются для логического разделения информации, привлечения внимания пользователя, отображения дополнительной информации. Для логического разделения информации можно использовать вкладки (tab). Вкладка позволяет в одном окне приложения переключение между несколькими открытыми документами или предопределенными наборами элементов интерфейса, когда их доступно несколько, а на выделенном для них пространстве окна можно показывать только один из них. Когда требуется привлечь внимание пользователя к какой-либо информации и (или) получить от него ответ, используются диалоговые

окна (dialog box). Еще одним применением дополнительных окон могут быть палитры элементов (toolbar). Палитры применяются, как правило, для размещения часто используемых функций.

Выполнение лабораторной работы

1 Изучить теоретическую часть.

2 Изучить возможности программных средств: Adobe Photoshop, Catalyst, Paint.NET, CorelDraw, Axure PR, www.wireframe.cc. Ознакомиться с элементами управления, позволяющими проектировать пользовательский интерфейс. Разработать описание элементов управления, предоставляемых данными программными средствами. Результаты представить в форме таблицы 6.1

3 Выбрать элементы управления для реализации полей, перечисленных в задании. Обосновать выбор элементов управления. Заполнить таблицу 6.2.

4 Разработать интерфейс программы.

5 Подготовить отчет.

6 Защитить лабораторную работу

Таблица 6.1 – Элементы управления, предоставляемые программными средствами

Программное средство	Элементы управления и их краткая характеристика
1	2
1 Adobe Photoshop	1 2 3 4 5
2 Catalyst	1 2 3 4 5
3 Paint.NET	1 2 3 4 5

Продолжение таблицы 6.1

1	2
4 CorelDraw	1 2 3 4 5
5 Axure PR	1 2 3 4 5
6 www.wireframe.cc	1 2 3 4 5

Таблица 6.2 – Обоснование выбранных элементов управления для выполнения задания лабораторной работы

Выбранный элемент управления	Обоснование

Варианты заданий для выполнения лабораторной работы

Вариант 1. Программа для добавления, сортировки и просмотра информации о фильмах:

- поле ввода названия фильма;
- поле подтверждения ввода;
- поля для отображения различной информации о фильме (название, год, режиссер и т. д.);
- поле для отображения всех добавленных фильмов;

- поля сортировки фильмов по различным критериям (название, год, режиссер и т. д.);
- поле плаката фильма.

Вариант 2. Программа для воспроизведения аудиофайлов:

- поле уровня громкости (с возможностью регулировки);
- поле текущей позиции воспроизведения (с возможностью изменения);
- поле для отображения названия файла;
- поле меню по управлению работы с файлом (воспроизведение, пауза, остановка, перемотка);
- поле главного меню для открытия файлов.

Вариант 3. Программа для работы с текстом:

- поле с текстом (с возможностью редактирования);
- поле для отображения различной информации о файле (название, количество строк);
- поле меню по изменению шрифта (тип, размер, цвет);
- поле главного меню для открытия и сохранения файлов;
- поле номера текущей страницы (с возможностью изменения).

Вариант 4. Программа для просмотра изображений:

- поле изображения;
- поле переключения между изображениями;
- поле меню для работы с текущим изображением (увеличение, изменение контраста);
- поле отображения текущего масштаба;
- поле названия текущего файла.

Вариант 5. Программа для просмотра текущих запущенных процессов и потребляемых ими ресурсов:

- поле запущенных процессов (с возможностью выбора одного из них);
- поле общей загруженности оперативной памяти и процессора;
- поле потребляемой памяти одним процессом (отображается при выборе процесса);
- поле остановки процесса;
- поле запуска нового процесса (с возможностью ввода названия нового процесса).

Вариант 6. Программа для обмена сообщениями:

- поле списка контактов (с разделением контактов на группы);
- поле текста отправляемого сообщения;
- поле текста получаемого сообщения;
- поле подтверждения отправки;
- поле для отображения текущего выбранного контакта (с возможностью отображения фотографии).

Вариант 7. Файловый менеджер:

- поле выбора папки для отображения ее файлов (отображение всех папок в виде иерархии);
- поле для отображения файлов (названия);
- поле подробной информации по выбранному файлу (размер, дата создания);
- поле удаления выбранного файла;
- поле поиска файла по названию.

Вариант 8. Почтовый клиент:

- поле папок писем (входящие, спам, удаленные);
- поле писем из выбранной папки;
- поле текста выбранного письма;
- поле информации о письме (отправитель, тема);
- поле удаления писем (с возможностью удаления нескольких выбранных).

Вариант 9. Программа для просмотра всех подключенных к компьютеру устройств:

- поле списка групп устройств (сетевые адаптеры, жесткие диски и т. д.);
- поле списка устройств из выбранной группы;
- поле поиска устройства по названию;
- поле включения/выключения устройства;
- поле дополнительной информации о выбранном устройстве.

Вариант 10. Словарь:

- поле списка доступных языков;
- поле ввода слова для поиска в словаре;

- поле отображения вариантов перевода (с возможностью удаления/добавления вариантов);
- поле алфавитного указателя.

Содержание отчета

- 1 Цель работы.
- 2 Краткие теоретические сведения.
- 3 Таблица с перечнем элементов управления, предоставляемых программными средствами Adobe Photoshop, Catalyst, Paint.NET, CorelDraw, Axure PR, www.wireframe.cc.
- 4 Таблица с обоснованием выбранных элементов управления.
- 5 Шаги разработки интерфейса в скриншотах.
- 6 Разработанный интерфейс программы.
- 7 Выводы по работе.

Контрольные вопросы

- 1 Дайте определение понятию элемент управления.
- 2 Какие правила расположения элементов управления необходимо соблюдать?
- 3 На какие типы делятся элементы управления по форме?
- 4 Охарактеризуйте каждый тип элемента управления по форме.
- 5 Перечислите набор элементов управления, относящихся к группе кнопок.
- 6 Перечислите набор элементов управления, относящихся к группе списков.
- 7 Перечислите набор элементов управления, относящихся к группе поля.

Лабораторная работа №7

ИССЛЕДОВАНИЕ ЭРГНОМИЧНОСТИ ИНФОРМАЦИОННОЙ СИСТЕМЫ

Цель работы: формирование у студентов знаний и практических навыков создания контрольных списков и проведения оценки юзабилити, создания персонажей для оценки юзабилити, оценки юзабилити веб-сайта и составления рекомендаций по улучшению.

Теоретические сведения

Критерий юзабилити демонстрирует степень эффективности, с которой продукт может использоваться пользователями для достижения определенных задач в определенном контексте. Основными критериями юзабилити являются: эффективность; продуктивность; удовлетворенность пользователя.

Эффективность продукта определяется как точность и полнота, с которой пользователи достигают поставленных целей. Эффективность измеряется следующими параметрами.

1 Процент задач пользователя, реализуемый в продукте (тот продукт будет считаться более эффективным, который реализует больше задач пользователей).

2 Отношение числа успешных действий к ошибкам (чем выше этот показатель, т. е. чем меньше ошибок совершается пользователем, тем эффективность продукта выше).

3 Нагрузка на пользователя (должна быть минимальной). Здесь прежде всего следует учитывать нагрузку на основные психические процессы: память, внимание, воображение. Интерфейс веб-сайта должен быть запоминающимся не только по внешним характеристикам, но и по принципам работы в нем. Интерфейс сайта должен быть интуитивно понятным пользователю.

4 Количество используемых функций и команд (должно быть высоким). Далеко не все функции и возможности, заложенные в продукте, используются пользователем. Если он реализует хотя бы 10 % из них, то такой продукт можно считать успешным.

Продуктивность определяется ресурсами, которые пользователь затрачивает, чтобы с точностью и полнотой достигнуть поставленных целей, и измеряется следующими показателями.

1 Время, которое необходимо пользователю для выполнения задания (должно быть минимальным).

2 Время, необходимое для предварительного обучения (должно быть минимальным). Этот показатель важен для веб-сайта – продукта, рассчитанного на широкую аудиторию. Чтобы минимизировать время для предварительного обучения пользователей, необходимо использовать стандарты, которые уже существуют в индустрии. Прежде всего целесообразно использовать те навыки, которые пользователи получили на других сайтах.

3 Время, которое тратится на ошибки и решение проблем. Большинство действий, которые пользователь совершает на сайтах, являются ненамеренными. В любой ситуации, в которой оказался пользователь, он должен иметь возможность легкого выхода из нее.

4 Количество совершаемых ошибок.

5 Частота использования справки и документации. Частота обращений к справке должна быть минимальной или, по крайней мере, не повторяющейся. Если пользователь один раз обратился за помощью, то это еще не является критерием низкого юзабилити. Если же он раз за разом перечитывает, не может запомнить предоставленные в справке указания, то на это стоит обратить внимание и проанализировать, почему это происходит.

6 Количество повторных и ошибочных действий. Если пользователь ошибается один раз – это нормально, он не должен ошибаться постоянно, особенно после предварительного обучения.

Удовлетворенность – комфорт и приемлемость использования продукта пользователем. Определить удовлетворенность достаточно сложно, поскольку это прежде всего эмоциональная компонента. Были введены следующие оценки: рейтинговая оценка по шкале полезности продукта или услуги; рейтинговая оценка по шкале удовлетворенности функциональностью продукта; количество случаев, когда пользователь испытывал гнев или проявлял недовольство при обращении с продуктом; оценка того, насколько технологическое выполнение задачи соответствует потребностям пользователя.

Юзабилити-тестирование – это эксперимент, выполняемый с целью определения того, насколько хорошо люди могут использовать некий искусственный объект для его предполагаемого применения. Юзабилити-тестирование – это метод оценки удобства использования продукта, основанный на привлечении пользователей в качестве тестирующих.

В ходе юзабилити-экспертизы выполняются следующие виды работ:

1) экспертная оценка основных юзабилити-характеристик продукта;

2) анализ существующего продукта на наличие проблем и недостатков

уровня:

- информационной архитектуры – совокупность методов и приемов структурирования информации);
 - пользовательского интерфейса и взаимодействия;
 - графического оформления, дизайна, верстки;
- 3) ранжирование проблем и недостатков по степени важности, сложности и т. д.

Исследование и оценка информационных систем может проводиться разными методами. Общим для всех методов является постановка реальных задач перед пользователями, а также фиксирование результатов тестирования для дальнейшего анализа. Юзабилити-тестирование позволяет выявить, отвечает ли продукт поставленным целям. Для участия в юзабилити-тестировании отбираются пользователи, соответствующие целевой аудитории. Они не должны быть слишком знакомы с разработкой информационной системы, чтобы не имели возможности быстрее других разобраться с ее устройством, что может создать иллюзию, что система понятна и удобна для целевых посетителей. Для того чтобы выявить и оценить наибольшее количество присутствующих проблем, необходимо привлекать реальных пользователей будущей информационной системы, а не своих коллег по работе.

Проблемы юзабилити предлагается оценивать по следующим четырем факторам: частота возникновения той или иной проблемы у разных пользователей; влияние проблемы на пользователей – сложна она или проста в решении; частота возникновения данной проблемы у одного пользователя; получается ли у пользователя решить проблему с первого раза или он сталкивается с ней каждый раз, когда она возникает.

Выделяют следующие виды юзабилити-тестирования: карточная сортировка (Card Sorting); контекстное исследование (Contextual Inquiry); контрольные листы (Checklists); макетирование (Prototyping); обзоры (Surveys); опросники (Questionnaires); плюралистическая проработка (Pluralistic Walkthroughs); протоколы самоотчета (Self-Reporting Logs); фиксация «мыслей вслух» (Thinking Aloud Protocol); фокусные группы (Focus Groups); эвристическое исследование (Heuristic Evaluation); экспертиза компонентов (Feature Inspection).

Карточная сортировка – это классификационный метод, при котором пользователи сортируют различные элементы разрабатываемого веб-сайта по нескольким категориям. Для проведения карточной сортировки создается список параметров, которые предполагается подвергнуть классификации, после чего каждый из указанных параметров записывается на отдельную карточку. Карточки предъявляются пользователям, которых инструктируют, как их сгруппировать наиболее логичным, по их мнению, образом. Полученную в результате

карточкой сортировки информацию используют для организации пользовательского интерфейса.

Контекстное исследование – это метод структурированного интервью, которое отличается от обычного, например журналистского, интервью тем, что оно всегда построено на трех базовых принципах: учет контекста, в котором используется изучаемый сайт; совместная оценка сайта пользователем и разработчиком; в фокусе оценки сайта находится именно его удобство для пользователя.

Контекстное исследование является одной из альтернатив метода эталонного тестирования, при котором удобство оценивается в лабораторных условиях, а не в привычной для пользователя рабочей обстановке. При контекстном исследовании работа, время, мотивация и социальные факторы, воздействующие на пользователя, остаются такими же, как в реальном мире, в отличие от лабораторных исследований, где эти факторы контролируются экспериментатором. Контекстное исследование наиболее применимо для того, чтобы оценить ту обстановку, в которой будет использоваться продукт, поэтому оно проводится на ранних стадиях его разработки. Наиболее предпочтительно использование контекстного исследования совместно с эталонным тестированием, так как каждый из этих методов по отдельности менее эффективен, чем их совокупность.

Контрольные листы помогают удостовериться в том, что веб-сайт выполнен с учетом принципов функциональности дизайна. Обычно их используют на заключительной стадии работы в дополнение к экспертным методам для того, чтобы структурировать экспертные оценки по каким-то определенным признакам. Существует большое количество готовых контрольных листов, однако решение об использовании того или иного списка должно зависеть от задач исследования. Зачастую возникает необходимость в разработке собственных критериев качества в определенной области.

Макетирование – это создание модели конечного продукта (веб-сайта), позволяющее протестировать его составляющие на любых стадиях разработки. В процессе макетирования строится модель, включающая все тестируемые компоненты (дизайн, элементы управления и т. д.). Можно использовать различные способы ее построения от изображения элементов интерфейса на бумаге до создания рабочего макета веб-сайта. Различают горизонтальное и вертикальное макетирование.

Горизонтальное макетирование означает представление в макете широкого спектра параметров веб-сайта, но без их детальной проработки. Этот вид макетирования предпочтительно использовать для изучения пользовательских предпочтений в области интерфейса. В процессе исследования выясняют, где, по мнению пользователей, должны находиться те или иные функции, насколько

они доступны и т. д. Горизонтальное макетирование применяют на ранних стадиях разработки.

При вертикальном макетировании исследуют функциональность отдельных компонентов веб-сайта. Например, при «вертикальном» макетировании сетевой базы данных могут изучаться ее поисковые возможности, но проверка ссылок, скорость загрузки, работа в разных браузерах и т. д. при этом не проводится. Так как этот вид макетирования предполагает детальное изучение небольшого сектора веб-сайта, его следует проводить лишь после завершения разработки тестируемого сектора.

Обзоры – это специальные интервью с пользователями, при которых им задаются специально подготовленные вопросы, а их ответы записываются для дальнейшей обработки. Традиционная методология проведения обзоров может также играть важную роль в исследовании веб-сайтов. Вопросы, входящие в обзор, могут варьироваться в зависимости от целей исследования, но, как правило, группируются по следующим категориям: демографическая информация о пользователе (место проживания, возраст, род занятий и т. д.); оценка информационного наполнения веб-сайта (какую информацию ожидали найти, соответствует ли наполнение веб-сайта их ожиданиям, что хотели бы добавить и т. д.); оценка дизайна сайта (легкость в использовании, скорость загрузки, оформление и т. д.).

Обзоры используют как на стадиях концептуализации и разработки для проведения маркетинговых исследований, идентификации потенциальных пользователей, установления их информационных нужд и компетентности в использовании компьютеров, так и после реализации веб-сайта для оценки реакций пользователей на информационное наполнение и удобство. Огромное количество образцов и результатов обзоров доступно через любую поисковую систему по ключевым словам «web usability» и «survey».

Опросники для оценки веб-сайтов составляются по тем же принципам, что и психодиагностические тесты. Их основное отличие от обзоров состоит в форме представления пользователям. Если обзоры – это структурированные интервью, то опросники предъявляются в виде списка вопросов с вариантами ответа.

Существует достаточно много стандартизированных опросников, большинство из которых распространяется на коммерческой основе, однако западные опросники нуждаются не только в переводе, но и в кросс-культурной адаптации. В зависимости от того, какие вопросы входят в опросник, этот метод может применяться на всех стадиях разработки веб-сайта или в процессе его эксплуатации.

Плюралистическая проработка проводится большой по размеру группой, в которую помимо экспертов в области оценки веб-продукции входят пользователи, разработчики и специалисты по профессиональному здоровью и организации труда.

Группа шаг за шагом рассматривает сценарии поведения пользователя, обсуждая и прорабатывая каждый элемент его взаимодействия с веб-сайтом. Вовлечение в эксперимент специалистов различных специальностей обеспечивает оценку веб-сайта с самых разнообразных позиций, а их замечания способствуют идентификации имеющихся недостатков. Метод применяется на ранних стадиях разработки в целях получения обратной связи как от специалистов, так и от конечных пользователей.

Протоколы самоотчета – это бланки типа «карандаш – бумага», в которых пользователи фиксируют все свои действия и соображения о взаимодействии с веб-сайтом. Этот метод является достаточно экономным, так как специалисты вовлекаются лишь в обработку результатов, не контролируя действия пользователя во время выполнения заданий.

Основными недостатками этого метода, как и других методов, предполагающих самостоятельную работу пользователей, является невозможность контроля и регистрации их эмоциональных реакций на взаимодействие с веб-сайтом и проблема адекватности их отчетов тому, что они реально делают. Поэтому в данном случае отбор пользователей, участвующих в эксперименте, приобретает принципиальный характер. Для проведения исследования необходимо обеспечить испытуемых доступом к прототипу веб-сайта, описанием задач, которые они должны решать при его использовании, и стандартизированным бланком для регистрации пользователями своих действий. Обычно эта техника используется на ранних стадиях планирования или разработки, а также для выявления пользовательских предпочтений.

Фиксация мыслей пользователя, вовлеченного в эксперимент, является одной из самых популярных техник при оценке функциональности веб-сайта. Пользователя просят произносить вслух все мысли, чувства и представления, которые у него возникают в процессе решения задачи.

Пользователя обеспечивают доступом к тестируемому веб-сайту или его прототипу и дают ему задание, которое он должен реализовать в процессе его эксплуатации. Он должен выполнять задачу, одновременно «озвучивая» все, что приходит в голову по поводу интерфейса. Данные записываются на аудиопленку или фиксируются письменно.

В отличие от большинства других методов эта техника позволяет оценить непосредственные реакции пользователя на взаимодействие с отдельными компонентами веб-сайта, не отсроченные по времени. И если его ожидания в отношении необходимых для решения задачи операций расходятся с дизайнерским решением веб-сайта, возможно, следует изменить это решение.

Несмотря на то что основной задачей техники является выяснение пользовательских представлений, с ее помощью можно реализовывать и другие цели. Например, терминология, которую употребляет пользователь для обозначения тех или иных элементов интерфейса, может быть использована и в дизайне веб-сайта.

Метод предполагает обобщение данных, полученных от нескольких пользователей. Существует также близкий к этой технике метод ответов на вопросы, использующий не вербализацию мыслей и эмоций, а директивные вопросы экспериментатора с фиксацией ответов пользователя.

Метод фокусных групп заключается в опросе специально отобранной группы пользователей. В исследование, которое обычно продолжается около 2 ч, вовлекается от 6 до 9 пользователей. Основное достоинство фокусных групп состоит в том, что они позволяют выявлять спонтанные реакции и идеи и оценивать отношение к этим идеям группы в целом.

Как правило, участники группы воспринимают происходящее как относительно свободный неструктурированный процесс, но ведущий группы должен иметь предварительный сценарий работы, вытекающий из целей исследования, и следить, чтобы групповая дискуссия не выходила из русла обсуждаемой проблемы. Кроме того, необходимо добиваться равного участия в дискуссии всех членов группы. Достаточно часто бывает, что в группе выделяется лидер, доминирующий над остальными участниками. Такие ситуации мешают свободному выражению мыслей и идей остальных членов группы и также нуждаются в коррекции. Результаты работы фокусной группы заносятся в специальный протокол для дальнейшей обработки. Сбор детальной информации при этом методе затруднен из-за относительной стихийности группового процесса, поэтому рекомендуется проводить несколько фокусных групп, состоящих из репрезентативных пользователей.

Несмотря на ряд преимуществ фокусных групп, они имеют и свои недостатки. Главным из них является неточность оценки, основанной на утверждениях, мыслях и предпочтениях небольшого количества пользователей. Поэтому при оценке веб-сайта фокусные группы должны использоваться лишь наряду с другими методами. Фокусные группы могут использоваться как на любой стадии разработки веб-сайта, так и для оценки готового продукта.

Эвристическое исследование проводится группой из 4–6 профессионалов в области экспертных оценок веб-продукции и взаимоотношений человека и компьютерных систем. Их работа базируется на соотношении качества веб-сайта со специально сформулированными эвристическими принципами. Каждый из экспериментаторов тщательно изучает веб-сайт, работая изолированно от других членов группы и письменно фиксируя результаты. Это необходимо для того, чтобы обеспечить независимость оценки каждого из экспериментаторов. Далее проводится групповое обсуждение полученных данных и вырабатывается коллективное решение. Результаты деятельности группы суммируются руководителем исследования.

Метод может применяться как при разработке макета веб-сайта, так и на поздних стадиях его изготовления, а также при анализе готового продукта.

При проведении юзабилити-тестирования сайтов следует учитывать следующие аспекты: тестируются возможности сайта, а не способности пользователей; необходимо обращать внимание на то, как пользователи выполняют задания, а не на то, как они оценивают сайт, чему отдают предпочтение; результаты юзабилити-тестирования будут являться основой для рекомендаций по улучшению сайта; необходимо искать наилучшее решение проблем, выделенных разными пользователями.

Выполнение лабораторной работы

- 1 Изучить теоретическую часть.
- 2 Разработать 20 юзабилити-требований к тестируемой информационной системе.
- 3 Определить функциональную единицу информационной системы для тестирования. Обосновать сделанный выбор.
- 4 Разработать список из 10-ти заданий для проведения юзабилити-тестирования пользователя, которые необходимо выполнить в процессе тестирования функциональной единицы информационной системы. Для каждого задания установить максимальное время выполнения. Заполнить таблицу 7.1.
- 5 Составить 10 вопросов о качестве продукта, который должен заполнить пользователь после прохождения тестирования (таблица 7.2).
- 6 Для проведения юзабилити-тестирования привлечь 10 пользователей, информацию о которых представить в виде таблицы 7.3.
- 7 Провести дважды юзабилити-тестирование с интервалом не менее 20 мин. Результаты представить в виде таблицы 7.4.

8 На основании результатов юзабилити-тестирования и опросника выявить проблемные места информационной системы. Сделать общую оценку системы. Разработать три прототипа с решениями выявленных проблем.

9 Подготовить отчет.

10 Защитить лабораторную работу.

Таблица 7.1 – Задания для проведения юзабилити-тестирования

Задание для проведения тестирования	Максимальное время выполнения задания
1	
2	
3	
4	
5	
6	
7	
8	
9	
10	

Таблица 7.2 – Опросник для испытуемых о качестве продукта

Вопросы о качестве продукта	Ответы пользователя
1	
2	
3	
4	
5	
6	
7	
8	
9	
10	

Таблица 7.3 – Информация о пользователях, принимающих участие
в юзабилити-тестировании

ФИО	Возраст	Образование	Род деятельности
1			
2			
3			
4			
5			
6			
7			
8			
9			
10			

Таблица 7.4 – Результаты юзабилити-тестирования

ФИО испытуемого	Номер экспе- римента	Время, затраченное на выполнение задания									
		t ₁	t ₂	t ₃	t ₄	t ₅	t ₆	t ₇	t ₈	t ₉	t ₁₀
1	1										
	2										
2											
3											
4											
5											
6											
7											
8											
9											
10											

Варианты заданий для выполнения лабораторной работы

- Вариант 1. <http://www.tut.by/>.
- Вариант 2. <http://www.dev.by/>.
- Вариант 3. <http://www.360.by/>.
- Вариант 4. <http://www.open.by/>.
- Вариант 5. <http://www.shara.by/>.
- Вариант 6. <http://www.relax.by/>.
- Вариант 7. <http://www.it-job.by/>.
- Вариант 8. <http://www.dodivana.by/>.
- Вариант 9. <http://www.onliner.by/>.
- Вариант 10. <http://www.euroopt.by/>.
- Вариант 11. <http://www.mts.by/>.
- Вариант 12. <http://www.bsu.by/>.
- Вариант 13. <http://www.gismeteo.by/>.
- Вариант 14. <http://www.velcom.by/>.
- Вариант 15. <http://www.belarusopera.by/>.

Содержание отчета

- 1 Цель работы.
- 2 Краткие теоретические сведения.
- 3 Краткая характеристика информационной системы.
- 4 Перечень требований к тестируемой системе.
- 5 Обоснование выбора функциональной единицы информационной системы для тестирования.
- 6 Список заданий для проведения юзабилити-тестирования пользователя.
- 7 Опросник о качестве продукта.
- 8 Информация об участниках юзабилити-тестирования.
- 9 Результаты юзабилити-тестирования.
- 10 Описание выявленных проблем. Общая оценка информационной системы. Разработанные прототипы с решениями выявленных проблем.
- 11 Выводы по работе.

Контрольные вопросы

- 1 Дайте определение понятию юзабилити.
- 2 Какими параметрами измеряется эффективность?

- 3 Какими показателями измеряется продуктивность?
- 4 Дайте определение понятию юзабилити-тестирование.
- 5 Какие виды работ выполняются в ходе юзабилити-экспертизы?
- 6 Перечислите методы, используемые для исследования и оценки сайтов.
- 7 Охарактеризуйте метод юзабилити-тестирования *карточная сортировка*.
- 8 Охарактеризуйте метод юзабилити-тестирования *контекстное исследование*.
- 9 Охарактеризуйте метод юзабилити-тестирования *контрольные листы*.
- 10 Охарактеризуйте метод юзабилити-тестирования *макетирование*.
- 11 Охарактеризуйте метод юзабилити-тестирования *обзор*.
- 12 Охарактеризуйте метод юзабилити-тестирования *мысли вслух*.
- 13 Охарактеризуйте метод юзабилити-тестирования *фокусные группы*.

Библиотека БГУИР

Лабораторная работа №8

ИССЛЕДОВАНИЕ ИНФОРМАЦИОННОЙ СИСТЕМЫ С ИСПОЛЬЗОВАНИЕМ СИСТЕМ ВИДЕООКУЛОГРАФИИ

Цель работы: формирование у студентов практических навыков проведения юзабилити-исследования информационных систем и использования результатов окулографии в юзабилити-исследованиях.

Теоретические сведения

Юзабилити-тестирование пользовательского интерфейса является важной задачей при его проектировании. Начальный этап тестирования связан с разработкой прототипа пользовательского интерфейса. На этом этапе используются имеющиеся результаты проектирования: общая схема приложения, планы отдельных экранных форм, глоссарий. Эти результаты сводятся воедино в общую схему, которую необходимо проверить по сформулированным сценариям.

Целью такой проверки является выявление несоответствий последовательности действий, описанных в сценарии, и структуры полной схемы. Обнаруженные несоответствия должны быть устранены за счет модификации экранных форм и/или корректировки общей схемы приложения.

Успех юзабилити-тестирования зависит от правильности и корректности постановки задач тестирования. Тестирование может быть направлено на подтверждение:

1 Производительности действий при использовании продукта. Оценивается по длительности выполнения задач (тестовых заданий) пользователем. Эффективный продукт позволяет увеличить число пользователей успешно выполняющих задание в течение ограниченного времени.

2 Полезности продукта. Продукт является полезным, если позволяет снизить количество человеческих ошибок.

3 Простоты обучения. Оценивается по времени тренинга, необходимого для достижения пользователем определенного уровня владения продуктом.

4 Субъективной оценки пользователей. Пользователи оценивают свое отношение к продукту по десятибалльной шкале.

Продукт можно считать успешным, если определенная часть пользователей оценила его на 8 и выше баллов. Тестовые задания, которые в ходе проведения представляют собой задачи для пользователей, формируются исходя из указанных задач тестирования. Основой формулирования тестовых заданий являются пользовательские сценарии. В тестировании участвуют представители

пользовательской аудитории, ранее не знакомые с разрабатываемым продуктом. Уровень опытности тестируемых пользователей должен соответствовать уровню, определенному в профилях конечных пользователей. Считается, что тестирование с участием одного пользователя позволяет выявить примерно 60 % ошибок. Поэтому число тестируемых пользователей, необходимых для проведения одного сеанса, зависит от сложности и объема проектируемого продукта. Для приложений средней сложности достаточно 4–8 человек.

В ходе тестирования:

- нельзя прерывать или смущать пользователя;
- нельзя внушать тестируемому, что тестируют его;
- желательно присутствие разработчиков приложения (программистов),

но их роль в тестировании исключительно пассивная.

В качестве методов проведения тестирования могут быть использованы следующие.

1 Наблюдение за пользователем. Пользователю предъявляется тестовое задание, он его выполняет. Действия пользователя фиксируются. Этот метод эффективен при определении неоднозначности элементов пользовательского интерфейса: любая неоднозначность, как правило, влечет за собой ошибку пользователя. Поскольку действия пользователя фиксируются, обнаружить ошибки при анализе тестов несложно. Кроме того, этот метод подходит для оценки производительности действий пользователя. Для этого необходимо при фиксировании действий замерять время, потребовавшееся пользователю на его выполнение.

2 Комментарии пользователя. Пользователь выполняет тестовые задания. Действия пользователя также фиксируются, кроме того, фиксируются комментарии его действий. В дальнейшем комментарии позволяют выявить недостатки реализации конкретных элементов пользовательского интерфейса – неудачное расположение элементов управления, плохая навигация и т. д. Этот метод можно использовать для оценки полезности продукта, простоты обучения работе с ним, степени субъективного удовлетворения. Следует отметить, что метод является «нестабильным»: результаты его использования зависят от личных качеств тестируемого пользователя – его разговорчивости, умения последовательно и внятно излагать свои мысли.

3 Качество восприятия. Пользователю предъявляется тестовое задание, через некоторое время после его выполнения, пользователь должен воспроизвести экранные формы (бумажный вариант), с которыми он работал. Результат воспроизведения сравнивают с оригиналом. Идея теста заключается в следующем. Из-за ограничения объема кратковременной памяти количество элементов экранных

форм, которые запоминает тестируемый, не может быть выше порога запоминания. Пользователь запоминает только то, что считает наиболее актуальным в процессе работы. Следовательно, при повторном выполнении задания пользователю, знающему расположение необходимых для этого элементов пользовательского интерфейса, будет проще. Таким образом, этот метод позволяет оценить простоту обучения работе с продуктом, а кроме того, степень субъективной удовлетворенности пользователей. Следует отметить, что выявление в ходе тестирования различных ошибок и несоответствий неизбежно. Это является одной из причин того, что тестирование нельзя переносить на окончание проекта, когда вносить модификации нет возможности из-за истечения сроков работ. Разработка пользовательского интерфейса приложения представляет собой итеративный процесс. Каждая итерация связана с отдельным этапом проектирования, созданием прототипа по его результатам, тестированием прототипа и его модификацией. Разработчик должен прилагать особые усилия, чтобы уменьшить число итераций.

При проведении юзабилити-исследований используют системы видеоокулографии (ай-трекеры, трекеры глаз), назначение которых заключается в регистрации направления взгляда человека при его взаимодействии с различными объектами. Параметрами таких систем являются: точность измерений (минимальное изменение углового перемещения взгляда); частота сканирования (количество кадров в секунду, которое делает камера системы); допустимое расстояние до объекта (расстояние пользователя от исследуемого объекта); допустимый диапазон движения головы (диапазон линейных или угловых перемещений вправо/влево и вверх/вниз, которые пользователь может совершать в ходе тестирования без потери данных); монокулярный или бинокулярный способ регистрации (регистрация направления взгляда от одного либо двух глаз).

К основным функциям систем видеоокулографии можно отнести следующие:

- построение зрительных маршрутов (траекторий перемещения взгляда человека по объекту) – можно наблюдать как в ходе тестирования, так и при воспроизведении его результатов;

- построение зон внимания или тепловых карт – позволяет оценить зоны фиксации взгляда на наблюдаемом объекте;

- статистическая обработка параметров движений глаз относительно разных зон, произвольно выделяемых экспериментатором.

Выполнение лабораторной работы

1 Изучить теоретическую часть.

2 Выбрать информационную систему для проведения исследования.

Сформулировать задачи исследования.

3 Разработать список из пяти заданий для проведения исследования, заполнить таблицу 8.1.

4 Провести исследование информационной системы с использованием систем видеоокулографии (ай-трекера глаз) на пяти испытуемых, заполнить таблицу 8.2.

5 Для каждого задания исследования разработать таблицу 8.3, в которую занести значения следующих параметров:

– удовлетворенность (оценка своих субъективных ощущений пользователем от работы с интерфейсом);

– эффективность (выполнение – способность пользователя решить поставленную задачу в отведенное время);

– ошибки – количество ошибок, допущенных при выполнении задания;

– время – значение затраченного времени на выполнение задания в секундах.

6 Проанализировать полученные результаты, соотнести их количественные оценки с качеством информационной системы. Результаты анализа записать.

7 Зарегистрировать максимальное время фиксации взгляда в определенной точке с помощью видеоокулографии (ай-трекера глаз), заполнить таблицу 8.4.

8 Произвести расчет параметров с помощью калькулятора SUMv5.xls. Результаты представить в виде таблицы 8.5.

9 Сформировать рекомендации по дальнейшей модификации информационной системы.

10 Подготовить отчет по лабораторной работе.

Таблица 8.1 – Задания для проведения тестирования информационной системы с использованием системы видеоокулографии

Задание для проведения тестирования	Время выполнения задания
1	
2	
3	
4	
5	

Таблица 8.2 – Информация о пользователях, принимающих участие в исследованиях

ФИО	Возраст	Образование	Род деятельности
1			
2			
3			
4			
5			

Таблица 8.3 – Результаты исследования информационной системы с использованием системы видеоокулографии для задания 1

ФИО испытуемого	Удовлетворенность	Эффективность (выполнение)	Ошибки	Время
1				
2				
3				
4				
5				

Примечание

Удовлетворенность: шкала от 0 до 5 баллов.

Эффективность: 1 – задание выполнено; 0 – задание не выполнено.

Ошибки – количество ошибок, допущенных при выполнении задания.

Время – значение затраченного времени на выполнение задания в секундах.

Таблица 8.4 – Максимальное время фиксации взгляда испытуемого в определенной точке информационной системы, обоснование полученных данных

ФИО испытуемого	Максимальное время фиксации взгляда	Точка фиксации взгляда (скриншот)	Обоснование
1			
2			
3			
4			
5			

Таблица 8.5 – Результаты исследования информационной системы с использованием системы видеоокулографии

Задание	Удовлетворенность	Эффективность (выполнение)	Ошибки	Время
Задание 1				
Задание 2				
Задание 3				
Задание 4				
Задание 5				

**Варианты заданий для проведения исследования
(для студентов заочной формы обучения)**

Вариант 1

Макет сайта: <http://www.nano.ru>. На этом макете сайта необходимо выполнить следующее:

- 1) найти правила оформления статей;
- 2) найти содержание последнего выпуска;
- 3) найти содержание других статей;
- 4) найти статьи К. Д. Яшина;
- 5) найти и прочитать условия подписки;
- 6) перейти на сайт английской версии журнала.

Вариант 2

Макет сайта: <http://www.belgopatent.org.by/>. На этом макете сайта необходимо выполнить следующее:

- 1) узнать состав руководства;
- 2) узнать, какие услуги предлагает данный интернет-ресурс;
- 3) найти образовательные услуги в сфере интеллектуальной собственности;
- 4) выяснить, какие есть еще информационные ресурсы Интернета в сфере интеллектуальной собственности;
- 5) перейти на сайт евразийской патентной организации;
- 6) найти патент в области электроники.

Вариант 3

Макет сайта: <http://zdorovie.com/>. На этом макете сайта необходимо выполнить следующее:

- 1) узнать, какие рубрики есть в данном журнале;

- 2) выбрать новую статью из актуальных;
- 3) найти на сайте подписку на рассылку;
- 4) найти для себя интересную статью;
- 5) найти, как поместить рекламу на сайте;
- 6) перейти на английскую версию сайта.

Вариант 4

Макет сайта: <http://infotechlib.narod.ru/>. На этом макете сайта необходимо выполнить следующее:

- 1) найти пять книг популярных авторов;
- 2) найти самые новые книги;
- 3) найти книги в области надежности и диагностики;
- 4) прочесть последние новости;
- 5) выяснить, сколько разделов в данной библиотеке;
- 6) узнать, что еще есть на сайте.

Вариант 5

Макет сайта: <http://www.gazeta.ru/>. На этом макете сайта необходимо выполнить следующее:

- 1) прочесть последнюю статью;
- 2) найти статьи в области техники;
- 3) выяснить, сколько рубрик есть в этой газете;
- 4) найти любую статью через поисковую систему сайта;
- 5) узнать, кто состоит в редакции журнала;
- 6) найти информацию о стоимости размещения информации в новостях.

Вариант 6

Макет сайта: <http://topsto-crimea.ru/>. На этом макете сайта необходимо выполнить следующее:

- 1) найти пять самых дорогих телевизоров;
- 2) найти пять самых мощных ноутбуков (по характеристикам);
- 3) узнать о гарантии;
- 4) выяснить, какие способы доставки предлагает данный интернет-магазин;
- 5) узнать, какие виды товаров предлагает интернет-магазин;
- 6) выяснить, какие способы оплаты предлагает данный интернет-магазин.

Содержание отчета

- 1 Цель работы.
- 2 Краткие теоретические сведения.
- 3 Краткое описание информационной системы. Задачи исследования информационной системы.
- 4 Задания для проведения исследования (см. таблицу 8.1).
- 5 Данные об испытуемых (см. таблицу 8.2).
- 6 Результаты проведенного исследования (см. таблицу 8.3).
- 7 Анализ полученных результатов, сопоставление количественной оценки с качеством продукта.
- 8 Данные о максимальном времени фиксации взгляда в определенной точке с помощью видеоокулографии (см. таблицу 8.4).
- 9 Расчет параметров по результатам исследования с помощью калькулятора SUMv5.xls (см. таблицу 8.5).
- 10 Рекомендации по дальнейшей модификации сайта.
- 11 Выводы по работе.

Контрольные вопросы

- 1 Какие задачи способно решать юзабилити-тестирование?
- 2 Как необходимо осуществлять подбор пользователей, которые будут принимать участие в тестировании информационной системы?
- 3 В чем заключается суть метода «качество восприятия»?
- 4 Какие параметры системы видеоокулографии необходимо учитывать при проведении исследований?
- 5 Перечислите основные функции систем видеоокулографии.

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

- 1 Лоранжер, Х. Веб-дизайн: удобство использования веб-сайтов / Х. Лоранжер, Я. Нильсен. – М. : Вильямс, 2007. – 368 с.
- 2 Нильсен, Я. Веб-дизайн / Я. Нильсен. – СПб. : Символ-Плюс, 2007. – 512 с.
- 3 Ортони, Э. Эмоциональная реакция и конструирование компьютеров / Э. Ортони, Д. Рассел, Д. Норман // Открытые системы. – 2003. – №10. – С. 26–29.
- 4 Буч, Г. UML / Г. Буч, Дж. Рамбо, А. Якобсон. – 2-е изд. – СПб. : Питер, 2006. – 736 с.
- 5 Буч, Г. Язык UML. Руководство пользователя / Г. Буч, Дж. Рамбо, А. Якобсон ; пер. с англ. Н. Мухина. – 2-е изд. – М. : ДМК-Пресс, 2007. – 496 с.
- 6 Информатика. Энциклопедический словарь для начинающих / сост. Д. А. Поспелов. – М. : Педагогика-Пресс, 1994. – 352 с.
- 7 Вайнштейн, Л. А. Эргономика : учеб. пособие / Л. А. Вайнштейн. – Минск : ГУИСТ БГУ, 2010. – 399 с.
- 8 Вайнштейн, Л. А. Психология труда: курс лекций / Л. А. Вайнштейн. – Минск : БГУ, 2008. – 220 с.
- 9 Человеко-машинное взаимодействие: теория и практика : учеб. пособие / О. С. Логунова [и др.]. – Ростов н/Д : Феникс, 2006. – 285 с.
- 10 Основы инженерной психологии / Б. А. Душков [и др.]. – М. : Академический Проект ; Екатеринбург : Деловая книга, 2002. – 576 с.
- 11 Вильямс, Р. Дизайн для недизайнеров / Р. Вильямс. – СПб. : Символ-Плюс, 2008. – 192 с.
- 12 Гарретт, Д. Веб-дизайн: книга Джесса Гарретта. Элементы опыта взаимодействия / Д. Гарретт. – СПб. : Символ-Плюс, 2008. – С. 192.
- 13 Раскин, Д. Интерфейс: новые направления в проектировании компьютерных систем / Д. Раскин ; пер. с англ. – СПб. : Символ-Плюс, 2004. – 272 с.
- 14 Купер, А. Психбольница в руках пациентов, или Почему высокие технологии сводят нас с ума и как восстановить душевное равновесие / А. Купер. – СПб. : Символ-Плюс, 2004. – 336 с.
- 15 Морвиль, П. Информационная архитектура в Интернете / П. Морвиль, Л. Розенфельд. – 3-е изд. – СПб. : Символ-Плюс, 2010. – 608 с.
- 16 Соев, С. Проектируем время. Психология восприятия времени в программном обеспечении / С. Соев. – СПб. : Символ-Плюс, 2009. – 224 с.

17 Желязны, Дж. Говори на языке диаграмм: пособие по визуальным коммуникациям / Дж. Желязны ; пер. с англ. – 6-е изд. – М. : Манн, Иванов и Фербер, 2016. – 304 с.

18 Research-Based Web Design & Usability Guidelines. – Washington : U.S. Department of Health and Human Services, 2005. – 267 p.

19 Валенко, В. С. Электроника и микросхемотехника : учеб. пособие / В. С. Валенко. – Минск : Бестпринт, 2003. – 319 с.

20 Семантическая модель сложноструктурированных баз данных и баз знаний : учеб. пособие по курсу «Модели представления знаний, базы данных и системы управления базами данных» для студентов специальности «Искусственный интеллект» всех форм обучения / В. В. Голенков [и др.]. – Минск : БГУИР, 2004. – 263 с.

21 Зенкин, А. А. Когнитивная компьютерная графика / А. А. Зенкин. – М. : Наука. – 1991. – 192 с.

22 ISO 18529. Эргономика человеко-компьютерного взаимодействия.

23 ISO 14915. Эргономика программного обеспечения мультимедийных пользовательских интерфейсов.

24 ISO 16071. Эргономика взаимодействия «человек – система».

Библиотека БГУИР

Учебное издание

**Калилец Татьяна Валерьевна
Осипович Виталий Семенович
Киринович Ирина Федоровна и др.**

ЭРГОНОМИКА ИНФОРМАЦИОННЫХ СИСТЕМ

ПОСОБИЕ

Редактор *Е. И. Герман*
Корректор *Е. Н. Батурчик*
Компьютерная правка, оригинал-макет *О. И. Толкач*

Подписано в печать 14.04.2018. Формат 60x84 1/16. Бумага офсетная. Гарнитура «Таймс».
Отпечатано на ризографе. Усл. печ. л. 4,53. Уч.-изд. л. 4,5. Тираж 50 экз. Заказ 23.

Издатель и полиграфическое исполнение: учреждение образования
«Белорусский государственный университет информатики и радиоэлектроники».
Свидетельство о государственной регистрации издателя, изготовителя,
распространителя печатных изданий №1/238 от 24.03.2014,
№2/113 от 07.04.2014, №3/615 от 07.04.2014.
ЛП №02330/264 от 14.04.2014.
220013, Минск, П. Бровки, 6