

МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ
РОССИЙСКОЙ ФЕДЕРАЦИИ

САНКТ-ПЕТЕРБУРГСКИЙ НАЦИОНАЛЬНЫЙ
ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ ИНФОРМАЦИОННЫХ
ТЕХНОЛОГИЙ, МЕХАНИКИ И ОПТИКИ

С. Ф. Сергеев

**МЕТОДЫ ТЕСТИРОВАНИЯ И
ОПТИМИЗАЦИИ ИНТЕРФЕЙСОВ
ИНФОРМАЦИОННЫХ СИСТЕМ**

Учебное пособие



Санкт-Петербург

2013

Сергеев С. Ф. Методы тестирования и оптимизации интерфейсов информационных систем: учебное пособие. – СПб: НИУ ИТМО, 2013. – 117 с.

Рассматриваются теоретические и прикладные вопросы юзабилити-тестирования программных интерфейсов для терминалов системы электронного правительства. Описаны конкретные процедуры и программно-методические средства для юзабилити-тестирования программных интерфейсов. Даны рекомендации в области юзабилити для проектировщиков программных интерфейсов.

Издание адресовано студентам магистерской программы «Управление государственными информационными системами» и слушателям дополнительной образовательной программы повышения квалификации «Электронное правительство и инновационные технологии управления», реализуемой Центром технологий электронного правительства НИУ ИТМО.

Рекомендовано к печати учёным советом Магистерского корпоративного факультета.



В 2009 г. университет стал победителем многоэтапного конкурса, в результате которого определены 12 ведущих университетов России, которым присвоена категория «Национальный исследовательский университет». Министерством образования и науки Российской Федерации была утверждена программа развития государственного образовательного учреждения высшего профессионального образования «Санкт-Петербургский государственный университет информационных технологий, механики и оптики» на 2009–2018 г.г.

© Санкт-Петербургский национальный исследовательский университет информационных технологий, механики и оптики, 2013

© С. Ф. Сергеев, 2013

Оглавление

Введение.....	6
Глава 1. Интерфейс. Общие понятия и структура.....	9
1.1. Интерфейс. Базовые определения	9
1.2. Основные определения и виды программного пользовательского интерфейса ...	10
1.3. Пользовательский интерфейс. Функции. Структура	11
1.3.1. Основные функции и требования к пользовательскому интерфейсу.....	14
1.3.2. Структура пользовательского интерфейса	16
1.4. Контрольные вопросы	17
1.5. Темы для дискуссии.....	18
Глава 2. Понятие «юзабилити» и основные этапы юзабилити-тестирования.....	19
2.1. Основные понятия юзабилити. Юзабилити-тестирование	19
2.2. Полное и промежуточное тестирование	24
2.3. Проведение промежуточного юзабилити-тестирования	24
2.4. Участие проектировщика в юзабилити-исследовании	25
2.5. Подготовка к тестированию.....	27
2.6. Проведение тестирования.....	28
2.7. Анализ полученных данных.....	29
2.8. Контрольные вопросы	30
2.9. Темы для дискуссии.....	31
Глава 3. Юзабилити-тестирование интерфейсов.....	32
3.1. Определение юзабилити в ISO 9126.....	33
3.2. Юзабилити-тестирование на этапах проектирования интерфейсов	33
3.3. Процесс исследования и его участники	42
3.4. Исследование пользователей	44
3.4.1. Маркетинговые исследования.....	44
3.4.2. Исследование контекста	44
3.5. Контрольные вопросы	45
3.6. Темы для дискуссии.....	46
Глава 4. Основные методы юзабилити-тестирования программных интерфейсов	47
4.1. Анализ рабочих заданий.....	47
4.2. Сегментация пользовательской аудитории	48
4.3. Персонажи.....	48
4.4. Оценка производительности	49
4.5. Мысли вслух	50
4.6. Конструктивное взаимодействие.....	50
4.7. Фокус-группы	50
4.7.1. Процесс исследования и его участники	51
4.7.2. Особенности методики.....	52
4.8. Метод карточной сортировки	53

4.8.1. Процесс исследования и его участники	54
4.8.2. Особенности методики карточной сортировки	55
4.9. Экспертная оценка	56
4.10. Эвристическая оценка	56
4.10.1. Процесс исследования и его участники	57
4.10.2. Особенности методики	58
4.11. Макетирование (прототипирование)	59
4.12. Бумажное прототипирование	60
4.12.1. Процесс исследования и его участники	60
4.12.2. Особенности методики	61
4.13. Контрольные вопросы	62
4.14. Темы для дискуссии	63

Глава 5. Программно-аппаратные методы оценки юзабилити пользовательских интерфейсов

5.1. Принципы и методы регистрации движения глаз	64
5.2. Системы ай-трекинга	67
5.3. Основные понятия о работе трекеров движений глаз	70
5.4. Компьютерные программы для проведения юзабилити-исследований	71
5.4.1. Общие сведения о программе <i>Mogae</i>	72
5.4.2. Базовые характеристики систем обработки данных анализа движений глаз	75
5.5. Типовой состав лаборатории юзабилити пользовательских интерфейсов	76
5.6. Контрольные вопросы	80
5.7. Темы для дискуссии	81

Глава 6. Рекомендации по повышению юзабилити компьютерных пользовательских интерфейсов

6.1. Философия дизайна компьютерного интерфейса	82
6.1.1. Использование метафор	82
6.1.2. Явные и подразумеваемые действия	84
6.1.3. Прямая манипуляция	85
6.1.4. Обратная связь и коммуникации	86
6.1.5. Согласованность в интерфейсе	87
6.1.6. «Что вижу, то и получаю»	88
6.1.7. Обеспечение возможности ошибаться	88
6.1.8. Воспринимаемая устойчивость интерфейса	89
6.1.9. Эстетическая целостность	89
6.1.10. Совместимость интерфейсной среды с операционной системой	90
6.1.11. Соответствие функций интерфейса достигаемым целям	91
6.1.12. Простота в использовании	92
6.1.13. Качественная графика	93
6.1.14. Скорость отклика системы на действия пользователя	94
6.1.15. Предоставление альтернативных способов выполнения задач	95
6.1.16. Поддержка доступности команд приложения	95
6.1.17. Правильное использование стандартных элементов интерфейса	96
6.1.18. Принципы оказания помощи пользователю	96
6.1.19. Применение анимации	97
6.1.20. Обеспечение безопасного исследования приложения пользователем	98

6.1.21. Обеспечение адаптации пользователя к изменениям окружающей среды	99
6.1.22. Проектирование взаимодействия приложений	100
6.1.23. Поддержка расширений файлов	100
6.1.24. Обеспечение доступной пользователю помощи	100
6.1.25. Обеспечение удобного ввода данных	101
6.1.26. Использование терминологии, ориентированной на пользователей	101
6.2. Общие правила дизайна при создании мультимедийных интерфейсов	102
6.2.1. Правила шрифтового оформления	102
6.2.2. Правила выбора цветовой гаммы	102
6.2.3. Правила общей композиции	103
6.2.4. Общее стилевое оформление	103
6.2.5. Рекомендации по дизайну мультимедийного представления информации	103
6.2.6. Текстовая информация	104
6.2.7. Графическая информация	104
6.2.8. Эффекты анимации	104
6.2.9. Использование звука	104
6.2.10. Единое стилевое оформление	105
6.2.11. Содержание и расположение информационных блоков	105
6.2.12. Общие правила оформления текста	105
6.2.13. Специфические требования при компьютерном наборе текста	107
6.3. Контрольные вопросы	108
6.4. Темы для дискуссии	109

Рекомендуемая литература

Приложение 1. Список стандартов по проектированию пользовательских интерфейсов и юзабилити

Приложение 2. Список интернет-ресурсов по юзабилити

Введение

Инженерная психология и эргономика в настоящее время являются наиболее востребованными психологическими дисциплинами в практике проектирования систем и средств пользовательского интерфейса для широкого круга информационных систем массового назначения.

Особенно популярно направление развития данных дисциплин, оформившееся под названием «юзабилити». Оно объединяет классические методы инженерной психологии с рыночной идеологией и направлено на оценку пользовательских качеств интерфейсов. Интерес к методам юзабилити неслучаен. Он связан прежде всего с интенсивным развитием техногенной среды человечества, появлением Интернета и необходимостью создания компьютерных интерфейсов для массового пользователя.

Особенностью пользователей массовых информационных систем является отсутствие у них систематического образования в области компьютерных технологий и программирования и наличие специфического опыта, не связанного с практикой работы с существующими компьютерными интерфейсами. Следствием этого является отсутствие или слабое развитие у массового пользователя навыков работы с интерфейсами информационных и технических систем. При этом число задач, ориентированных на широкого пользователя, стремительно растет в связи с всеобщей компьютеризацией и тенденцией замены человеческого общения в некоторых профессиональных областях на общение с компьютерными системами. В разряд массовых пользователей попадают пользователи электронных терминалов по продаже услуг сотовой связи, оплаты коммунальных услуг, налогов и т. д.

Одной из важных задач является создание электронного правительства, замещающего функции чиновников муниципального звена и обеспечивающего осуществление массовых государственных услуг, связанных с взаимоотношениями с налоговыми органами, органами УВД и т. д. Именно здесь наблюдается наибольшее число конфликтных ситуаций, связанных с непосредственным общением с чиновниками. Для решения задач обеспечения функционирования электронного правительства необходимо создание систем программного интерфейса, учитывающих свойства массовых пользователей и не требующих специализированного обучения. Системы должны быть просты и доступны при сохранении своей функциональности для всех категорий пользователей. Именно эту задачу решает технология юзабилити-тестирования, изложению которой посвящено настоящее учебное пособие.

За рубежом практика привлечения юзабилити-специалистов к созданию и оценке различных аспектов программных продуктов, связанных с взаимодействием, существует уже достаточно давно, но лишь в последнее десятилетие приобрела массовый характер.

В Советском Союзе методы инженерной психологии и эргономики при проектировании интерфейсов традиционно использовали в оборонной промышленности при создании массовых видов вооружений. В связи с существовавшей тогда завесой секретности данные приложения мало известны широкой публике, и в силу этого юзабилити воспринимается в современной практике создания информационных систем достаточно новой дисциплиной, не имеющей глубоких научных традиций и корней.

Это — ошибочное мнение. Научно-практические основы юзабилити широко представлены в работах отечественных и зарубежных специалистов в области инженерной психологии и эргономики (Т. П. Зинченко, В. М. Львов, Б. Ф. Ломов, В. М. Магазаник, П. И. Падерно, П. Я. Шлаен и др.). Однако на практике больший интерес представляет облегченная, адаптированная к массовому проектировщику систем интерфейсов, версия инженерной психологии в виде технологий юзабилити, излагаемых в книгах и руководствах «гуру» — Я. Нильсена, А. Купера, Д. Раскина, Р. Унгера, С. Круга и др.

С каждым годом внимание к юзабилити информационных продуктов возрастает. Все большее количество разработчиков программных средств задумываются над тем, как сделать работу со своими продуктами максимально доступной, комфортной и эффективной. Растет понимание того, что юзабилити может стать серьезным конкурентным преимуществом в борьбе за потребителя.

Технологии и типовые методы инженерной психологии и юзабилити, применяемые в соответствующих ситуациях проектирования, должны быть известны практическим работникам. Все это определяет необходимость подготовки специалистов, владеющих методами юзабилити программных интерфейсов информационных систем.

Цель настоящего учебного пособия состоит в формировании системы знаний студентов в области оценки и проектирования пользовательских интерфейсов информационных систем методами классической инженерной психологии и юзабилити-тестирования.

Основная часть информации, используемой в данном пособии, была получена из работ исследователей, занимающихся вопросами человеко-машинного взаимодействия в сфере информационных технологий и эргатических систем. Их перечисление заняло бы не одну страницу, что не входит в цели настоящей работы.

Отметим ведущую роль в развитии современной отечественной версии юзабилити В. Н. Андреева, А. С. Баканова, А. Н. Костина, В. М. Львова, В. Д. Магазанника., Н. А. Назаренко, А. А. Обознова, П. И. Падерно, Д. К. Сатина, Е. Е. Сугак, Г. Л. Коротеева, П. Я. Шлаена и др.

В конце каждого раздела пособия приведены контрольные вопросы и темы для дискуссии. Они предназначены для самоконтроля при самостоятельной работе студентов и для отработки тем при групповом обучении.

Список принятых сокращений

АРМ — автоматизированное рабочее место;
КГР — кожно-гальваническая реакция;
МЭГ — магнитоэнцефалография;
ОС — операционная система;
ПИ — пользовательский интерфейс;
ПК — персональный компьютер;
ПО — программное обеспечение;
ПП — программный продукт;
ПЭТ — позитронно-эмиссионная томография;
СОИ — средства отображения информации;
ОУ — органы управления;
фМРТ — функциональная магнитно-резонансная томография;
ЭКГ — электрокардиография;
ЭЭГ — электроэнцефалография;
GOMS (Goals, Operators, Methods, and Selection rules) — модель целей, объектов, методов и правил выбора для эргономической оценки пользовательских интерфейсов;
НED — наголовные системы трекинга.

Глава 1. Интерфейс. Общие понятия и структура

1.1. Интерфейс. Базовые определения

Понятие «интерфейс» широко используется в технике и технологиях, обозначая комплекс средств и правил взаимодействия между отдельными элементами одной системы. Так, например, в компьютерной технике под интерфейсом понимается форма и технология общения между средствами ввода и программными средствами машины. В программировании интерфейс — это детализированная, описанная граница взаимодействия между кодом, предоставляющим некоторые возможности, и кодом, который эти возможности использует. Это — правила взаимодействия операционной системы с пользователями. Инженеры-схемотехники понимают под интерфейсом разъемы и кабели, с помощью которых соединяются между собой блоки электронной аппаратуры.

В более широком случае *интерфейс* — это часть системы, обеспечивающая взаимодействие с пользователем или другой системой. *Интерфейс человеко-машинный* — комплекс технических и информационно-программных средств, посредством которых осуществляется диалоговый режим взаимодействия человека-оператора и вычислительных средств компьютеризированного образца.

Программные средства интерфейса — совокупность программных средств, обеспечивающих диалог оператора с вычислительными средствами и визуализацию виртуальных объектов на экране.

Технические средства интерфейса — средства отображения информации (СОИ) и органы управления (ОУ), используемые оператором при осуществлении диалога с вычислительными средствами.

В рамках теории обучающих и профессиональных иммерсивных сред под интерфейсом понимается связь между средами, позволяющая действующим в них системам оказывать те или иные формы не разрушающих друг друга ориентаций, ведущих к диалогу, кооперации или другим формам взаимодействий¹.

Понятие *интерфейса* является контекстно-зависимым, и дать его точное однозначное определение на все случаи жизни невозможно. Вместе с тем в каждой системе можно выделить и конкретизировать ее элементы, выполняющие интерфейсные функции.

¹ Сергеев С. Ф. Обучающие и профессиональные иммерсивные среды. М.: Народное образование, 2009. – 432 с.

В самом общем виде интерфейс обозначает форму и реализацию связи между системой и другими системами, составляющими рабочую (окружающую) среду системы.

1.2. Основные определения и виды программного пользовательского интерфейса

С ростом и развитием сетевых систем передачи данных, таких как Интернет, широкое распространение получили интерфейсы, позволяющие обеспечить взаимодействие пользователя с различными прикладными и сервисными программами (через браузер, например), проводить мониторинг и управление заказом в интернет-магазине, отслеживать процессы движения грузов в службах экспресс-перевозок, настраивать и управлять удаленными системами и устройствами.

Программный интерфейс — система унифицированных связей, предназначенных для обмена информацией между компонентами вычислительной системы и оператором. Программный интерфейс задает набор необходимых процедур, их параметров и способов обращения.

Современными видами интерфейсов в системе «человек-компьютер» являются:

— *командный интерфейс*. Он называется так потому, что в этом виде интерфейса человек подает «команды» компьютеру, который их выполняет и выдает результат человеку. Командный интерфейс реализован в виде пакетной технологии и технологии командной строки;

— *WIMP-интерфейс* (*window* — окно, *image* — образ, *menu* — меню, *pointer* — указатель). Характерной особенностью этого вида интерфейса является то, что диалог с пользователем ведется с помощью графических образов — меню, окон, других элементов. Хотя и в этом интерфейсе подаются команды машине, но это делается «опосредованно», через графические образы информационной модели, реализованной на экране дисплея. Различают два вида реализаций этого интерфейса: простой графический и «чистый» WIMP-интерфейс;

— *SILK-интерфейс* (*speech* — речь, *image* — образ, *language* — язык, *knowledge* — знание). Этот вид интерфейса наиболее приближен к обычной, человеческой форме общения. В рамках этого интерфейса идет обычный «разговор» человека и компьютера. При этом компьютер формирует для себя команды, анализируя ключевые слова и выражения человеческой речи. Результат выполнения команд также преобразуется в понятную человеку форму;

— *веб-интерфейс* — это совокупность средств, при помощи которых пользователь взаимодействует с веб-сайтом через веб-приложение. Он

относится к программным интерфейсам. Веб-интерфейсы удобны тем, что дают возможность вести совместную работу сотрудникам, не находящимся в одном офисе (например, веб-интерфейсы часто используются для заполнения различных форм и баз данных или публикации материалов в интернете и средствах массовой информации).

В настоящее время появляются новые виды интерфейса, использующие биометрические показатели и поведенческие невербальные каналы коммуникации, анализируемые компьютером. Для управления компьютером используются жесты, мимика пользователя, направление его взгляда, размер зрачка и другие признаки. При идентификации пользователя используются характеристики радужной оболочки его глаз, отпечатки пальцев и другая информация. Изображения считываются с цифровой видеокамеры, а затем с помощью специальных программ распознаются и включаются в процедуры управления и диалога с компьютером.

Широко развиваются системы *семантического интерфейса*. Этот вид интерфейса стимулируется развитием исследований в области искусственного интеллекта. Он представляет собой симбиоз известных форм интерфейсов, объединенных семантическим анализом речевых сообщений. Его трудно назвать самостоятельным видом интерфейса — он включает в себя и интерфейс командной строки, и графический, и речевой, и мимический интерфейс. Основная его отличительная черта — использование естественного языка вместо языка команд. Моделируется общение. Запрос формируется на естественном языке, в процессе «коммуникации» с машиной.

1.3. Пользовательский интерфейс. Функции. Структура

Пользовательский интерфейс устанавливает правила для элементов интерфейса и интерактивной технологии, позволяющие пользователю эффективно решать профессиональную задачу. Именно через удобный и интуитивно понятный интерфейс пользователь судит о прикладной программе в целом, принимает решение об ее отклонении или использовании. Пользовательский интерфейс (рис. 1) предназначен для обеспечения взаимодействия между пользователем и процессом, выполняющим некоторое задание, — прикладной программой. Задачами данного взаимодействия являются передача информации от пользователя прикладной программе, а также результатов работы программы пользователю.



Рис. 1. Пользовательский интерфейс *Windows Vista*

Интерфейс также играет основную роль в интерпретации результатов работы прикладной программы. Современные операционные системы предоставляют пользователям широкие возможности по гибкому изменению вида и содержания пользовательского интерфейса. Рассмотрим, например, полнофункциональную версию *Windows Vista*, предлагающую четыре типа пользовательского интерфейса — упрощенный, классический, стандартный или *Windows Aero*.

Упрощенный интерфейс типичен для настольных ПК начального уровня, хотя, по сравнению с предыдущими версиями *Windows*, модернизирован даже он. Упрощенная версия интерфейса обладает новым меню «Пуск» для быстрого поиска и запуска любого приложения. Модернизированы проводники и такие вспомогательные функции, как интегрированный с рабочим столом поиск, масштабируемые иконки. Упрощенный интерфейс может использоваться с любой версией *Windows Vista* (в том числе *Starter*), он наименее требователен к оборудованию.

Классический интерфейс — это сочетание всех возможностей упрощенного интерфейса с оформлением в стиле *Windows 2000*. Этот вариант также включен во все выпуски *Vista*, в том числе в *Vista Starter*.

Стандартный интерфейс Windows Vista предназначен для настольных ПК среднего уровня и недоступен в *Vista Starter*. Это — типичный интерфейс ПК под *Vista Home Basic* с поддержкой *Windows Driver Display Model (WDDM)*. Стандартный интерфейс предъявляет такие же требования к оборудованию, как и интерфейс *Windows Aero*.

Графический интерфейс Windows Aero — это современный улучшенный интерфейс для ПК с графикой, поддерживающей технологию *WDDM*. Отличительной особенностью интерфейса *Aero* является профессиональный дизайн в стиле прозрачного стекла с различными эффектами, вроде отражения и плавной анимации. Прозрачные

«стеклянные» окна помогают сосредоточиться на содержимом, не отвлекаясь на окружающий интерфейс. Интерфейс *Aero* доступен только в выпусках *Windows Vista Home Premium, Business, Enterprise* и *Ultimate*. К числу основных особенностей интерфейса *Vista Aero* относятся следующие:

- «стекло» — окна *Aero* полностью прозрачны, возможна настройка цвета, насыщенности и даже уровня прозрачности в соответствии с предпочтениями пользователя;

- динамические окна — свёрнутое окно незаметно перемещается в указанное место на панели задач, позволяя быстро найти его в случае необходимости;

- масштабируемые эскизы на панели задач — отображают содержимое открытых и свернутых окон. Задержав указатель мыши над кнопкой на панели задач можно в реальном времени просмотреть содержимое соответствующего окна, не выводя его на передний план. Благодаря технологии *Windows Flip 3D*, если в этом свёрнутом окне происходит какое-то динамичное явление (крутится видеоролик, анимация, телепередача), предварительный просмотр в небольшом всплывающем окошке также покажет действие в реальном времени;

- *Windows Flip (Alt+Tab)*. По аналогии с *Windows XP* в версиях *Vista* имеется комбинация клавиш *Alt+Tab* для перемещения между окнами, но, если при включенном интерфейсе *Aero* листать окна с помощью комбинации клавиш *Win + Tab (START+TAB Flip 3D)*, создаётся тот самый 3-мерный эффект, когда динамически отображаются все открытые окна, располагаясь одно над другим в трехмерном виде. Функция *Windows Flip* отображает эскизы открытых окон вместо обычных значков, и в это же время, благодаря функции *Flip 3D*, отображаются такие динамические процессы, как воспроизведение видео. Одно из открытых окон в режиме *Flip* всегда будет находиться на рабочем столе, при этом можно просто и быстро свернуть все открытые окна. Для плавного переключения между открытыми окнами и выбора нужного окна можно использовать клавиши управления курсором или колесо прокрутки мыши; при этом открытые окна скользят по экрану, когда пользователь перемещает их или изменяет их размер.

Продолжения интерфейсов, реализованных в *Windows Vista* и *Windows 7*, показывают тенденцию к созданию интерфейса пользователем под свои требования и привычки.

1.3.1. Основные функции и требования к пользовательскому интерфейсу

В настоящее время наиболее часто используют программный и пользовательский виды интерфейса, и при этом имеющиеся стандарты указывают на то, какими эти интерфейсы быть не должны, т. е. весьма незначительно сужают множество возможных вариантов построения интерфейса.

Исследования в данной области показывают, что любой пользовательский интерфейс (ПИ) должен обеспечивать выполнение следующих четырех основных функций:

1. Управление компьютером путем реализации действий оператора (пользователя): инициация, прерывание, отмена компьютерных процессов и т. п.

2. Ввод данных, осуществляемый оператором, и отклик системы.

3. Отображение данных, вводимых оператором, который может управлять этим процессом.

4. Поддержка оператора в процессе деятельности, осуществляемая по каналам обратной связи, в которых циркулирует информация об ошибочных или случайных (не по алгоритму) действиях оператора.

Эффективный ПИ должен обеспечивать возможность всестороннего использования возможностей оператора, технических и информационно-программных средств автоматизированных рабочих мест (АРМ), безошибочность и скорость действий оператора при применении ПИ по назначению. Кроме того, хорошо спроектированный ПИ должен обеспечивать комфортную деятельность оператора, исключая появление напряженности и стресса, снижение уровня психофизиологических и психологических характеристик, необходимых для эффективного выполнения профессиональной деятельности.

В соответствии с отечественными и зарубежными стандартами пользовательский интерфейс должен:

- способствовать быстрому освоению компьютеризированной техники оператором, формированию у него стереотипов деятельности;
- быть спроектирован таким образом, чтобы оператор вводил информацию естественным образом, не заботясь о ходе вычислительного процесса;
- удовлетворять рабочие потребности человека-оператора, а не обеспечивать процесс обработки данных. Его синтаксическая и визуальная структуры должны быть согласованы с ожидаемыми

оператором результатами расчета, требованиями решаемой вычислительной задачи и используемыми средствами ввода;

- содержать систему правил работы оператора, обеспечивающую легкое управление системой;
- все время работы находиться под контролем оператора, никакие действия последнего не должны приводить к тупиковой ситуации или зависанию программы;
- обеспечивать возможность исправления ошибок ввода, не требовать повторного ввода данных;
- обеспечивать обратную связь. Подсистемы справки должны обеспечивать оператора информацией, позволяющей ему управлять диалогом, распознавать и исправлять ошибки, а также определять последующие действия, входящие в алгоритм деятельности.

Пользовательский интерфейс предъявляет особые требования к информации:

- выдаваемая компьютером информация должна быть краткой, ясной, конкретной и понятной оператору;
- объем представляемой оператору информации должен быть согласован с объемом его оперативной памяти;
- в информации об ошибках следует делать акцент не на неправильные действия оператора, а на то, чем и каким образом можно исправить возникшие ошибки.

В информационной системе необходимо предусматривать использование четырех видов диалога: через меню, команды, манипуляции и диалог посредством заполнения форм. При решении каждой задачи пользователю должна быть представлена возможность использования не менее двух видов диалога. Критерием выбора вида диалога в ходе решения конкретной задачи является обеспечение заданных показателей безошибочности и быстродействия.

Разработка пользовательского интерфейса осуществляется с позиций «дружественного интерфейса». Дружественный интерфейс должен:

- быть интуитивно понятным пользователям;
- гибко изменять свои настройки и свойства под предпочтения пользователя;
- быть предсказуемым для пользователя и давать ему возможность контроля над свойствами управляемой системы;

- прощать пользователю ошибки и защищать систему от ошибок пользователя;
- создавать у пользователя образ программного продукта как единого целого;
- сохранять результаты и возможность пользователю вернуться в любую точку алгоритма;
- помогать пользователю справочными и иными материалами для обеспечения его безопасной и эффективной работы в системе.

Данные требования обеспечивают партнерское взаимодействие человека с управляемой машиной.

Интерфейс должен отвечать эргономическим требованиям, связанным с безопасностью для пользователя, учетом его антропометрических, биомеханических, психофизиологических и иных свойств, отражающих свойства человека как биосоциального существа, реализующего целенаправленную созидательную деятельность.

1.3.2. Структура пользовательского интерфейса

Пользовательский интерфейс содержит три основных компонента:

- 1) визуальное оформление, отвечающее за представление информации оператору;
- 2) функциональные возможности системы, включающие набор возможностей для эффективного выполнения профессиональной деятельности;
- 3) техники взаимодействия оператора с системой.

Однако разработчики программных продуктов рассматривают функциональность системы отдельно от ее пользовательского интерфейса и практически не рассматривают элементы взаимодействия пользователя и системы. При этом предполагается, что ПИ является своего рода дополнением к функциональности системы. Со своей стороны, пользователи программ, как правило, не разделяют функциональность и пользовательский интерфейс. Для пользователей именно ПИ является программой. Впечатление от взаимодействия с программным продуктом (ПП) формируется непосредственно от работы с интерфейсом.

Поэтапная разработка пользовательского интерфейса позволяет повысить эффективность программного продукта, уменьшить время обучения пользователей, снизить стоимость доработки системы после ее внедрения, а также полностью использовать заложенную в ПП функциональность.

Для эргономической оценки пользовательских интерфейсов используется модель *GOMS* (*Goals, Operators, Methods, and Selection rules*) — модель целей, объектов, методов и правил выбора. Она построена на оценке времени затрат оператора при выполнении тех или иных операций с помощью интерфейса. Разработчики модели *GOMS* заметили, что время, требующееся для выполнения какой-то задачи системой «пользователь — компьютер», является суммой всех временных интервалов, которые потребовались системе на выполнение последовательности элементарных действий, составляющих данную задачу.

1.4. Контрольные вопросы

1. Что такое «интерфейс»? Основные понятия.
2. Назовите признаки хорошо спроектированного пользовательского интерфейса.
3. С помощью каких средств реализуется пользовательский интерфейс?
4. Назовите основные функции, решаемые с помощью пользовательского интерфейса.
5. С помощью каких систем интерфейсов реализуется взаимодействие человека с компьютером?
6. Что входит в структуру пользовательского интерфейса?
7. Что такое «дружественный интерфейс» и цели его проектирования?
8. Требования к информации в пользовательских интерфейсах.
9. Что такое «функциональность пользовательского интерфейса»?
10. Чем отличается взаимодействие от действия и взаимодействия?
11. Что такое «программный интерфейс»? Назовите его основные свойства.
12. Опишите в требованиях модели *GOMS* работу человека с клавиатурой в двух различных текстовых редакторах, например *Open Office* и *Microsoft Word 2003*.
13. Какие задачи в обеспечении взаимодействия в системе «человек — машина» решаются при проектировании интерфейса?
14. Является ли манипулятор «мышь» интерфейсом системы «человек — компьютер»?
15. Опишите работу речевого командного интерфейса и интерфейса, работающего на естественном языке.
16. Приведите примеры реализации интерфейсов, использующих биометрические показатели.

17. Опишите основные пользовательские свойства интерфейса *Windows Aero*.

18. Опишите тенденции развития интерфейсов для мобильных устройств.

1.5. Темы для дискуссии

1. Спроектируйте идеальный интерфейс для сотового телефона.
2. Опишите свойства человеческого тела как системы интерфейса между психическим и физическим мирами.
3. Мир как интерфейс.
4. Этапы эволюции интерфейсов операционной системы *Windows*.

Глава 2.

Понятие «юзабилити» и основные этапы юзабилити-тестирования

2.1. Основные понятия юзабилити. Юзабилити-тестирование

Юзабилити (применимость) изделия означает, что люди, которые используют это изделие, могут быстро и легко ставить и выполнять собственные задачи, не замечая интерфейс и его особенности.

Это определение опирается на четыре положения: 1) юзабилити (применимость) означает необходимость сосредотачиваться на пользователях; 2) человек использует изделия, чтобы быть производительным; 3) пользователи — занятые люди, пробующие выполнять задачи; 4) именно пользователи решают, когда изделие удобно².

Юзабилити означает, что та или иная вещь работает хорошо, если человек средних способностей и опыта может ее использовать по назначению вне зависимости от того, является ли она *Веб-сайтом*, реактивным истребителем или вращающейся дверью³. Юзабилити начинается с философии — веры в то, что проектирование удовлетворяет потребности пользователя и сосредотачивается на создании качественного пользовательского опыта. Для достижения реальной цели юзабилити требуются определенные технологии и методы. Юзабилити учитывает круг лиц, использующих изделие, отвечает на вопросы, каковы их цели и потребности, выбирает правильные методы, чтобы решить проблему: «Насколько хорошо это изделие выполняет юзабилити требования пользователей?».

Важно понять, что применимость — не одномерное качество пользовательского интерфейса. Юзабилити имеет много компонентов и традиционно связано с пятью признаками: обучаемостью, эффективностью, запоминаемостью, ошибками, удовлетворенностью⁴.

Юзабилити — это измеримая характеристика, присутствующая в большей или меньшей степени во всех продуктах, которая описывает, как эффективно (фактически) пользователь может взаимодействовать с изделием.

² Janice (Ginny) Redish and Joseph Dumas. A Practical Guide to Usability Testing, 1999, p. 4.

³ Steve Krug. Don't Make Me Think, 2000, p. 5.

⁴ Jakob Nielsen. Usability Engineering, 1993, p. 26.

«Юзабилити относится к степени, до которой изделие может использоваться указанными пользователями, чтобы достигнуть указанных целей с эффективностью и удовлетворением в указанном контексте использования». — Международная организация по стандартизации ISO 9241-11.

«Ориентированный на человека проект характеризуется активным вовлечением пользователей и ясным пониманием пользователями требований задачи, соответствующим распределением функций между пользователями и технологией, повторением решений проекта, мультидисциплинарностью проекта». — Международная организация по стандартизации ISO 13407.

Деловые выгоды от добавления юзабилити к процессу обеспечения жизненного цикла изделия включают:

- увеличение производительности;
- увеличение продаж и доходов;
- уменьшение времени и затрат на обучение;
- уменьшение времени и затрат на развитие;
- уменьшение затрат на обслуживание;
- увеличение удовлетворенности клиента.

Часто проектировочные организации не отслеживают общий объем затрат на жизненный цикл проекта программного обеспечения тренажера. Первоначальный бюджет развития и графики исполнения известны, но часто не учитываются затраты на поддержку, сколько рабочего времени потрачено впустую на непригодных инструментах и другие факторы производства. Таким образом, проекты, которые имеют маленькие начальные затраты и короткие циклы реализации, часто вознаграждаются, даже если в результате создаются некачественные и неэффективные изделия.

Юзабилити-тестирование — это набор методов и инструментов, позволяющих измерить характеристики взаимодействия пользователя с продуктом для оценки уровня реализации пользовательских свойств (юзабилити) продукта.

В ходе юзабилити-тестирования изучается, насколько хорошо пользователи выполняют конкретные стандартные задачи и с какими проблемами они при этом сталкиваются. Результаты такого тестирования помогают выявить как проблемы, затрудняющие понимание и использование продукта, так и удачные решения. Юзабилити-тестирование осуществляется в специальных лабораториях, оснащенных специальным

оборудованием, обеспечивающим проведение тестирования и соблюдение психометрических требований к процедуре тестирования.

Для проведения юзабилити-тестирования требуется, чтобы предмет тестирования обладал относительной завершенностью и внутренней согласованностью. Смысл тестирования состоит в проверке качества интерфейса продукта, не зависимо от того, тестируется ли готовое программное обеспечение, минимально функциональный макет или же бумажный прототип. Следовательно, юзабилити-тестирование должно проводиться на поздних стадиях цикла проектирования, когда уже существует концепция продукта и имеется достаточное число деталей, чтобы можно было создать его макет или прототип.

Можно привести определенные доводы в пользу уместности юзабилити-тестирования на ранних стадиях переработки интерфейса существующего продукта. В ряде проектов эта процедура действительно позволяет выявить аспекты продукта, нуждающиеся в улучшении. Однако крупные недостатки продукта можно обнаружить посредством *качественных исследований*, и если бюджет проекта не позволяет выполнить более одного юзабилити-тестирования, то гораздо полезнее использовать его как средство проверки конкретных элементов нового дизайна, когда будет готово решение-кандидат.

Поскольку результаты пользовательского тестирования обычно измеримы и поддаются количественному выражению, юзабилити-исследования продукта особенно ценны при сравнении конкретных вариантов дизайна для отбора наиболее эффективного решения. Собранные в ходе юзабилити-тестирования отзывы потребителей наиболее полезны в случае, когда планируется проверить (либо усовершенствовать) механизмы взаимодействия или форму и реализацию определенных элементов продукта.

Аспекты продукта, для оценки которых юзабилити-тестирование особенно эффективно:

1. *Наименование.* Осмысленны ли названия разделов и подписи на кнопках? Возможно, какие-то из этих слов воспринимаются легче, чем другие?
2. *Архитектура.* Осмысленно ли информация разбита на категории? Расположены ли информационные элементы в тех местах, где их ожидают найти потребители?
3. *Первое знакомство и доступность.* Легко ли новые пользователи находят базовые элементы интерфейса? Понятны ли инструкции? Есть ли в них необходимость?

4. *Эффективность.* Могут ли потребители эффективно решать конкретные задачи? Ошибаются ли они? При выполнении каких шагов? Как часто?

Из вышесказанного видно, что юзабилити-тестирование сосредоточено преимущественно на оценке первого опыта использования продукта. Зачастую очень сложно и всегда трудоемко измерять эффективность решения при пятидесятом использовании продукта. Один из методов преодоления этих трудностей носит название *дневниковые исследования*: испытуемые ведут дневники с подробными записями о своем взаимодействии с продуктом.

Наконец, при проведении юзабилити-тестирования следует убедиться, что вы тестируете то, что можно измерить; что тестирование выстроено корректно; что результаты будут полезны для выявления проблем проектирования, и у вас есть ресурсы, необходимые для исправления этих проблем.

В ходе проектирования взаимодействия зачастую желательно оценить, насколько качественными являются те решения, которые появились на свет. Для этого требуется выйти за пределы персонажей и проверочных сценариев и предложить решения реальным пользователям. Делать это следует тогда, когда решение обрело достаточную детализацию, чтобы пользователи могли реагировать на него как на что-то вполне конкретное, но при этом есть еще достаточно времени, чтобы внести исправления, исходя из результатов тестирования.

Юзабилити-тесты и сеансы, направленные на получение отзывов пользователей, хорошо подходят для выявления крупных проблем с инфраструктурой взаимодействия и для улучшения таких вещей, как надписи на кнопках, порядок и приоритет действий. Они также жизненно важны для настройки таких аспектов поведения, как скорость прокрутки экрана в качестве реакции на поворот аппаратной ручки. К сожалению, сложно создать тест, оценивающий что-либо, кроме простоты освоения решения новичками. Существует ряд методов для оценки удобства продукта в использовании средним пользователем или экспертом, но такие методы отнимают много времени и дают в (лучшем случае) не очень точные результаты.

Существует множество способов проверить интерфейс на реальных пользователях. Это — неформальные сеансы обратной связи, в которых поясняются идеи и выслушиваются соображения пользователя, и более строгие методы *юзабилити-тестирования*, в котором пользователи решают определенный набор задач. Каждый подход имеет свои преимущества. Более неформальные методы могут применяться спонтанно и требуют меньшей подготовки. Минус такого подхода в том, что

проектировщик часто оказывается виновен в «подсказках свидетелю», если объясняет в побуждающей манере. В целом мы находим, что этот подход приемлем для технической аудитории, которая способна анализировать немногочисленные рисунки, представляющие интерфейс продукта. Он может быть достойной альтернативой юзабилити-тесту, если у команды проектировщиков нет времени для подготовки к формальному юзабилити-тестированию.

Если же времени достаточно, необходимо отдать предпочтение более формальному юзабилити-тестированию. Юзабилити-тесты определяют, насколько хорошо найденное решение позволяет пользователям решать свои задачи. Если область тестирования достаточно широка, вы узнаете также, насколько хорошо решение помогает пользователям в достижении конечных целей.

Чтобы не было недоразумений, уточним, что юзабилити-тестирование по сути своей есть средство анализа, а не синтеза. Оно не служит альтернативой проектированию взаимодействия и никогда не будет являться источником идей, приводящих к созданию привлекательного продукта. Скорее, это — метод оценки эффективности существующих идей и инструмент для их совершенствования.

Кроме этого, нужно учитывать, что юзабилити-тестирование — это не исследование пользовательской аудитории. Некоторые практики считают, что «тестирование» может включать исследование (интервью, анализ задач) и даже творческие упражнения вроде «проектирования с привлечением пользователей». Однако это — попытка свести в один вид деятельности различные потребности и шаги процесса проектирования.

Изучение пользовательской аудитории должно проводиться до генерации идей, а юзабилити-тестирование — после. Более того, известно, что когда проектные ограничения заставляют выбирать между этнографическими исследованиями и юзабилити-тестированием, то время, потраченное на исследование, дает больше возможностей для создания привлекательного продукта. Точно так же, если время и бюджет ограничены заказчиком, то время, потраченное на проектирование, приносит в процесс большую ценность, нежели тестирование. Лучше проводить больше времени, принимая взвешенные, основанные на серьезном исследовательском фундаменте решения в области проектирования, чем тестировать сырые решения, не пользуясь преимуществами прозрачных и удобных моделей пользователей, их потребностей и целей.

2.2. Полное и промежуточное тестирование

Якоб Нильсен различает *полное тестирование*, когда тестируются завершённые продукты, и *промежуточное тестирование*, проводимое в ходе проектирования как часть итерационного процесса.

Полное тестирование применяется для сравнения продуктов, для выявления проблем перед перепроектированием, для расследования причин возвратов продукта и определения источника запросов на обучение и поддержку. Общие исследования, как правило, проводятся независимыми профессионалами и подробно документируются. В некоторых случаях, особенно при конкурентном анализе, общие исследования строятся таким образом, чтобы получить количественные данные, которые можно проверить на статистическую значимость.

К сожалению, полное тестирование часто используется в составе процесса обеспечения качества — ближе к завершению разработки. На этой стадии обычно уже поздно вносить осмысленные изменения в проектные решения. Оценку пользовательского интерфейса следует проводить до того, как начнется создание кода (или, по крайней мере, достаточно рано, чтобы было время изменить реализацию сообразно внесенным дополнениям). Однако если требуется убедить заинтересованных лиц или программистов в том, что проблема юзабилити в имеющемся продукте явно присутствует, то ничто не заменит наблюдения за реальными пользователями, сражающимися с программным продуктом.

Промежуточное тестирование предназначено как раз для этого. Эти быстрые качественные тесты проводятся в ходе проектирования, обычно на этапе детализации. Продуманное и контролируемое промежуточное тестирование «открывает окно» в разум пользователя, позволяя проектировщикам видеть, как их целевая аудитория реагирует на информацию и инструменты, предоставленные им для решения задач. Хотя у полного тестирования, безусловно, есть своя область применения, но обычно оно относится к сфере информационной поддержки планирования жизненного цикла продукта. Данный вид тестирования позволяет проводить проверки юзабилити в ходе разработки, но потери в деньгах, времени, боевом духе команды могут на данном этапе дорого обойтись. Промежуточное тестирование проводится во благо проектирования в ходе процесса проектирования.

2.3. Проведение промежуточного юзабилити-тестирования

Существуют различные мнения относительно того, как правильно проводить юзабилити-тестирование и интерпретировать его результаты. К сожалению, многие подходы к юзабилити либо пытаются заменить методы

инженерного проектирования, либо имеют перекос в сторону количественных метрик и дают бесполезные данные вроде «общего времени выполнения задачи». Работа Кэролин Снайдер «*Paper Prototyping*» (Snyder, 2003) — хороший справочник по методам юзабилити-тестирования, совместимый с методами целеориентированного проектирования. Хотя в ней описаны далеко не все методы тестирования и не затронуты отношения между тестированием и проектированием, но достаточно подробно рассмотрены основы и даются некоторые относительно простые методики для юзабилити-тестирования.

Основные советы К. Снайдер по успешному промежуточному юзабилити-тестированию:

1. Проводите тестирование достаточно поздно, когда уже существует конкретное проектное решение, но и достаточно рано, чтобы можно было успеть скорректировать проект и реализацию.
2. Тестируйте задачи и аспекты опыта пользователя, связанные с конкретным продуктом.
3. Набирайте участников из аудитории целевых пользователей, используя персонажей в качестве фильтра.
4. Четко ставьте перед участниками задачи и просите при их решении размышлять вслух.
5. Дайте участникам возможность напрямую взаимодействовать с низкотехнологичным прототипом (исключением являются случаи, когда тестируется специализированное оборудование, а бумажный прототип не способен отразить нюансы взаимодействия).
6. Управляйте ходом сеансов, чтобы выявить проблемы и изучить причины их возникновения.
7. Минимизируйте предвзятость, введя в эксперимент модератора, который ранее не принимал участия в проекте.
8. Сосредоточьтесь на поведении и образе мысли участников.
9. По завершении тестов проведите разбор проведенного исследования вместе с наблюдателями, чтобы выявить причины наблюдавшихся проблем.
10. Вовлекайте в процесс исследования проектировщиков.

2.4. Участие проектировщика в юзабилити-исследовании

Отсутствие взаимопонимания между проектировщиком и пользователем является распространенной причиной проблем с юзабилити продукта. Персонажи помогают проектировщикам понять цели пользователей, их потребности и точку зрения, формируя фундамент

эффективных коммуникаций. Юзабилити-исследование, открывая дополнительное окно в мышление пользователя, позволяет проектировщикам увидеть, как доносятся их вербальные, визуальные и поведенческие решения, узнать намерения пользователей при взаимодействии со спроектированными ожидаемыми назначениями.

Проектировщики (или лица, принимающие решения по проектированию) — основные потребители результатов юзабилити-исследований. Редкие проектировщики способны моделировать сеанс тестирования, сохраняя нейтралитет, однако их вовлечение в планирование исследований и прямое наблюдение за сеансами тестирования, а также участие в анализе и решении проблем крайне важно для успеха исследования.

Проектировщиков важно привлекать к следующим видам деятельности:

- планированию исследований для формулирования вопросов, важных с точки зрения проектирования;
- определению критериев отбора участников с помощью персонажей и их атрибутов;
- использованию сценариев для выбора тестовых задач;
- наблюдению за сеансами тестирования;
- совместному анализу результатов исследований.

Подводя итоги вышесказанного, следует заметить, что как бы ни совершенны были соображения, приведшие к созданию интерфейса, всегда остается вероятность того, что интерфейс получится плохим, либо (что более вероятно) не таким хорошим, каким он мог бы быть. Необходимо иметь какие-либо подтверждения его работоспособности. Все, что для этого нужно, — это несколько пользователей средней квалификации, а также прототип (разумеется, при наличии основного бюджета можно развернуться: например, купить прибор, фиксирующий направление взгляда пользователя и длительность зрительных фиксаций).

Юзабилити-тестирование включает в себя проведение экспериментов для выявления информации, касающейся эргономики и дизайна ПИ. Самый адекватный перевод слова *usability*, пожалуй, будет «потребительские качества продукта». Существование идеологии юзабилити-тестирования — придать ПИ максимально высокие потребительские свойства (удобство, продуктивность, легкость понимания и обучения). Короче говоря — все во имя пользователя и на благо пользователя. Юзабилити-тестирование — стержень эргономической экспертизы ПИ.

Существует распространенное заблуждение, что с помощью тестирования можно решить все проблемы интерфейса. С помощью тестирования можно определить только слабые места интерфейса, но почти невозможно обнаружить сильные, поскольку они пользователями просто не замечаются, и совсем уж невозможно определить новые способы улучшения интерфейса.

Юзабилити-тестирование производится на протяжении всего цикла разработки ПИ. На ранних этапах разработки тестирование предыдущей версии или конкурирующих продуктов позволяет наметить контрольные точки, которые необходимо достигнуть в процессе разработки. В середине работы над проектом тестирование предоставляет обратную связь, сообщая места, где ПИ нуждается в улучшении. На заключительных этапах тестирование удостоверяет, что продукт соответствует (или не вполне соответствует) тем целям, для которых был спроектирован.

2.5. Подготовка к тестированию

Прежде всего, надо очень ясно поставить вопрос о цели тестирования, причем в терминах, позволяющих однозначно ответить, является ли полученный результат ответом на этот вопрос.

Перед тестированием надо отчетливо понимать характеристики потенциальных пользователей, которые и должны стать участниками тестирования. Это необходимо потому, что когда вы займетесь непосредственно подбором участников, очень важно будет знать, кто именно вам нужен: новички, эксперты или опытные пользователи, мужчины, женщины, или же вы нуждаетесь в представителях обоих полов; также важен возраст пользователей.

Следует определить структуру процесса тестирования, его композицию, описывающую процедуру проведения отдельных тестов; в каком порядке они будут выстроены, чтобы исключить из рассмотрения и дальнейшего анализа переменные, не представляющие интереса. Разрабатываются задания, которые будут предложены участникам тестирования. Эти задания должны быть основаны на тех задачах, которые пользователи решают с помощью вашего продукта в процессе его нормального использования. Следует указать все, что понадобится вам для того, чтобы определить сценарий теста: состояние автомата, машины или компьютера, экраны, документацию, другие средства помощи и подсказки, которые должны присутствовать. Также следует указать, каким образом определяется успешное завершение выполнения каждого задания.

Перед тестированием надо также определить, какой вспомогательный инструментарий нужен, и приобрести его. Инструментарий может включать в себя приборы и устройства, используемые в процессе

проведения тестов. Это, например, — видеокамеры для записи поведения пользователей; устройства записи того, что происходит на экранах мониторов; диктофоны и записывающая аудиоаппаратура для протоколирования вербального общения и записи вербальных протоколов; односторонние зеркала, позволяющие наблюдателям и экспериментатору оставаться невидимым для участников тестирования, и т.д. Можно собрать много полезной информации, пользуясь простыми любительскими видеокамерами, или вообще обойтись без видеозаписей.

Необходимо составить список пользователей, из которого будут подбираться участники для каждого теста. Количество пользователей должно быть достаточным, чтобы создать репрезентативную выборку в требуемой для теста пропорции характеристик пользователей (опыта, навыков, демографических характеристик и т. д.), поскольку в противном случае неучтенные факторы могут повлиять на полученные данные. Профили пользователей, которые вы определили ранее, помогут вам создать модель типового пользователя вашего продукта. Например, приборной доской реактивного истребителя могут пользоваться не только сами пилоты, но и обслуживающие рабочие, механики, инструкторы и диагностирующий персонал.

2.6. Проведение тестирования

Многие люди чувствуют себя неуютно, когда попадают в лабораторию, где им приходится выполнять задания, зная, что время выполнения замеряется и все их ошибки записываются для дальнейшего анализа, поэтому очень сложно сделать так, чтобы пользователь чувствовал себя комфортно и спокойно. Важно подчеркнуть, что тестированию подвергается ПИ, а не пользователи, и им не стоит чувствовать себя под социальным давлением. Обязательно следует поблагодарить пользователей за участие в тестах. Нужно объяснить участникам, что они могут в любой момент остановить тест или сделать перерыв, если это кому-либо потребуется.

Нередко тестирование требует от участников предварительного соглашения о неразглашении информации и подтверждения их согласия на производство записей. В качестве части этой «бумажной работы» вы можете провести участников через анкетирование для того, чтобы идентифицировать знание предметной области, их позиции и отношения, или чтобы получить еще немного информации о характеристиках пользователей.

При проведении тестирования главное — это наблюдать за ходом их мысли. Когда нет уверенности, что вам понятно, о чем они думают, можно прямо спросить у них об этом.

Самая главная задача тестирования — понять, какие ожидания возникают у пользователей на каждом шагу и насколько программа и ее интерфейс соответствуют этим ожиданиям. Если пользователь говорит: «Я не знаю, что делать дальше», — вам нужно спросить: «А как вы думаете, что бы вы могли сделать?» или «А что бы вы сделали, если бы были дома?» Когда пользователь уже готов щелкнуть мышью, можно спросить у него, что он ожидает увидеть. После того как он щелкнет мышью, узнайте, действительно ли результат оправдал его ожидание.

Трудность состоит в том, что нельзя отвлекать пользователей или оказывать на них влияние, но в то же время необходимо выяснить, что они в действительности думают (что, кстати, они и сами могут не до конца понимать).

После каждого тестирования полезно делать короткие записи о том, что вам запомнилось. Если вы не сделаете этого до начала следующего теста, то потом вам будет очень трудно вспомнить важные детали.

Пользователи часто подсказывают решения какой-либо проблемы. Очень часто такие подсказки наводят на идеи, о которых раньше думали, но по каким-то причинам отклонили их. Иногда случается, что в проект были внесены какие-то изменения (например, было решено применить другую технологию или сместились коммерческие приоритеты), и становится полезным тот метод или подход, который был отклонен вначале.

2.7. Анализ полученных данных

После того как задания выполнены и тестирование завершено, следует обсудить результаты теста с его участником. Это необходимо для того, чтобы получить дополнительную информацию, касающуюся того, о чем думал пользователь во время тестирования. Одним из путей анализа событий является их восстановление и обсуждение с участником тестирования. Кроме этого, можно просто спросить участника, что из случившегося во время теста показалось ему заслуживающим внимания.

После тестирования каждый наблюдатель и помощник должны не позднее следующего дня написать небольшой отчет об основных проблемах, которые были замечены во время тестирования, а также изложить свои мысли по поводу способов их устранения. Не надо сразу писать полных и развернутых отчетов. Пусть это напоминает, скорее, резюме, краткие заметки. В идеале все участники группы по разработке должны прочитать эти отчеты (или хотя бы пробежать их глазами), поэтому объем этих документов не должен превышать 1–2 страницы.

На собрании рабочей группы по итогам юзабилити-исследования следует обсудить два основных вопроса:

1. Каковы проблемы, с которыми сталкивались пользователи, и какие из них должны быть исправлены?
2. Каковы возможные решения для отображенных проблем?

При анализе полученных данных в первую очередь следует искать крупные проблемы. Найти такие проблемы проще, поскольку они становятся заметны еще при просмотре заметок, сделанных во время наблюдения за пользователями. Если каждый участник сталкивался с проблемами при использовании определенного пункта меню, очевидно, что дизайн этого пункта нуждается в пересмотре.

Данные, касающиеся производительности, такие (как частота ошибок и время выполнения заданий), оцениваются с помощью методов статистического анализа. Большая часть такого анализа сводится к нахождению среднего значения и стандартного (среднего квадратичного) отклонения, а также к проверке достоверности полученных различий.

Наблюдения за действиями пользователей и запись их самоотчетов как во время теста (используя метод записи «мысли вслух» или задавая вопросы), так и до или после теста проводятся с использованием анкет и опросных листов. Структура большинства таких опросников позволяет количественно оценить ответы испытуемых, используя численную шкалу. Полученные количественные данные могут анализироваться также с помощью стандартных статистических процедур корреляционного и факторного анализа, реализуемых в пакетах *SPSS*, *STATISTICA PRISM*, широко используемых в выборочных исследованиях.

2.8. Контрольные вопросы

1. Назовите признаки изделия с высоким юзабилити.
2. Назовите пять признаков юзабилити пользовательского интерфейса.
3. Проанализируйте определение юзабилити по *ISO 9241-11*.
4. На какие основные положения опирается сама возможность определения юзабилити?
5. На каких пользователей рассчитаны методы юзабилити?
6. Можно ли измерить характеристики юзабилити?
7. Что означает продукт, ориентированный на пользователя?
8. Какие деловые выгоды можно получить при добавлении юзабилити к процессу обеспечения жизненного цикла изделия?
9. Что такое «юзабилити-тестирование»?
10. Что позволяет получить юзабилити-тестирование?
11. С какой целью создаются юзабилити-лаборатории?

12. В чем смысл юзабилити-тестирования?
13. На каких стадиях цикла проектирования должно проводиться юзабилити-тестирование? Ваши аргументы.
14. Назовите аспекты продукта, для оценки которых эффективно юзабилити-тестирование.
15. Какие проблемы позволяют выявить процедуры юзабилити-тестирования?
16. Что такое «полное» и «промежуточное» юзабилити-тестирование?
17. Для чего применяются полное и промежуточное юзабилити-тестирование?
18. Дайте советы по успешному проведению промежуточного юзабилити-тестирования.
19. К каким видам юзабилити-деятельности важно привлекать проектировщиков интерфейса?
20. В чем состоит идеология юзабилити-тестирования?
21. Почему с помощью юзабилити-тестирования можно определить только слабые места интерфейса?
22. Что нужно сделать на стадии подготовки к тестированию?
23. Как обеспечить проведение тестирования? На какие моменты взаимодействия с испытуемыми необходимо обратить особое внимание?
24. С какой целью обсуждается тест с участниками после его проведения?
25. Какие вопросы обсуждаются на заседании рабочей группы после тестирования?
26. Какие методы статистической обработки применяются при анализе результатов юзабилити-тестирования?

2.9. Темы для дискуссии

1. Что такое «пользовательские качества» интерфейса, и как они зависят от опыта пользователя?
2. Как организовать эффективную процедуру юзабилити-тестирования сотового телефона?
3. Лаборатория юзабилити. Оснащение и назначение основных элементов.
4. Правила поведения персонала юзабилити-лаборатории в процессе юзабилити-тестирования.

Глава 3.

Юзабилити-тестирование интерфейсов

Usability-тестирование — это экспериментальный метод, построенный на интервьюировании пользователей по заранее составленному сценарию и направленный на выяснение того, как пользователи используют продукт.

Участников тестирования, также называемых «респондентами», отбирают и нанимают на основе специализированной анкеты, составленной так, чтобы отсеять людей, не входящих в целевую аудиторию пользователей продукта.

В процессе каждого интервью модератор (фасилитатор) просит респондентов выполнять задачи, решаемые с помощью тестируемого продукта.

Смысл *usability-тестирования* состоит в наблюдении и анализе проблем, с которыми сталкиваются пользователи, работая с продуктом. Целью тестирования является улучшение интерфейсов на базе опыта целевой аудитории. При этом создается финальная версия продукта, позволяющая обеспечить наиболее эффективное взаимодействие.

Наблюдение за пользователями позволяет найти способы улучшения дизайна и функциональности продукта, но бесстрастно наблюдать за этим процессом довольно сложно. Человеку присуще желание и потребность помогать окружающим, а не наблюдать за их ошибками и неудачами. Эффективный модератор должен быть способен держать респондента сфокусированным на задачах, не помогая ему при этом решать их.

По этой причине модератор и респондент должны находиться в разных помещениях, общаясь друг с другом посредством различных устройств связи. Эти помещения обычно разделяют односторонним зеркалом Гезелла, позволяющим модератору и другим наблюдателям видеть, как работает респондент, не находясь в зоне его внимания.

Под термином «юзабилити-тестирование» скрывается совокупность различных методов, применяемых для определения юзабилити-характеристик программных продуктов. У них есть одна общая особенность, которая и объединяет эти методики, — участниками такого тестирования являются обычные пользователи из числа тех, кто работает (или предположительно будет работать) с объектом тестирования. Это отличает их от экспертных оценок, где свое заключение по данному вопросу в той или иной форме дают юзабилити-специалисты. Во время юзабилити-тестирования пользователей просят выполнить определенные задания с помощью изучаемого изделия или продукта и тем или иным

способом стараются извлечь полезные сведения из наблюдений за их действиями.

3.1. Определение юзабилити в ISO 9126

Термин «юзабилити» часто используется для обозначения способности продукта быть легко используемым. Этот взгляд согласуется с определением юзабилити как качества программного обеспечения в *ISO/IEC 9126*: «набор свойств программного обеспечения, которые связаны с усилием, требуемым для использования и с индивидуальной оценкой такого использования определенной или подразумеваемой группой пользователей». При этом свойства, которыми продукт должен обладать для достижения высокого уровня юзабилити, зависят от особенностей пользователя, задачи и среды. Для продукта не существует внутреннего юзабилити, есть только способность быть используемым в определенном контексте. Юзабилити нельзя оценить, исследуя продукт изолированно, вне пользовательского контекста.

Юзабилити продукта может быть оценено путем анализа:

- *продукта и контекста использования.* Юзабилити может быть измерено путем оценки свойств продукта, необходимых для достижения высокого уровня юзабилити в определенном контексте. Такие свойства описываются в других частях *ISO 9241*, однако *ISO 9241* дает только часть руководства. Из множества возможных проектных решений, совместимых с *ISO 9241*, одни более пригодны к использованию, чем другие.

- *процесса взаимодействия.* Юзабилити может быть измерено путем моделирования взаимодействия между пользователем, выполняющим задачу, и продуктом, однако современные аналитические методы не дают точных оценок юзабилити. Поскольку взаимодействие — это динамический процесс, протекающий в сознании человека, он не может быть исследован непосредственно. Однако измерения умственных усилий и субъективного отношения являются важными косвенными показателями характеристик, вносящих вклад в юзабилити.

- *эффективности, экономичности* при использовании продукта в определенном контексте и *измерения удовлетворенности* пользователей продукта. Поскольку это — непосредственные показатели юзабилити, они являются конечным тестом. Если продукт более пригоден к использованию в определенном контексте, показатели юзабилити будут выше.

3.2. Юзабилити-тестирование на этапах проектирования интерфейсов

Практика юзабилити накопила большое количество методов тестирования, применяющихся на разных этапах юзабилити-

проектирования и исследования интерфейсов. В переводе Александра Качанова⁵ приведен список статей, описывающих методы и приемы, используемые в дизайне программных систем вообще и веб-сайтов в частности. В соответствии с ним весь процесс дизайна веб-интерфейса условно разбивается на 6 этапов. На каждом из этапов используются свои методы, а их результаты затем становятся отправной точкой для использования других методов. Есть некоторые методы, которые могут применяться на нескольких этапах.

1. Этап планирования и оценки

На данном этапе используются следующие методы:

- собрание участников проекта;
- анализ контекста использования;
- *ISO 13407: Human centered design processes for interactive systems*;
- *Usability Planning*;
- анализ конкурентов.

Цель данного этапа — обеспечить эффективное использование методов в процессе дизайна и разработки, обеспечить постоянное влияние этих методов на проектировщиков с самой ранней стадии проекта.

На данном этапе рекомендуется следующая последовательность применения методов тестирования:

- начать с собрания участников проекта;
- проанализировать контекст использования будущего продукта;
- создать план обеспечения юзабилити на основе стандарта *ISO 13407*;
- проанализировать конкурентов и выявить сильные и слабые стороны их продуктов.

2. Этап составления требований к проекту использует другие методы:

- опрос пользователей;
- беседы с пользователями;
- беседа в контексте;
- наблюдение за пользователями исследования на месте;
- контекст использования;

⁵ Перевод Александра Качанова «Методы веб-дизайна и юзабилити» [Электронный ресурс] <http://webmascon.com/topics/development/22a.asp> по материалам сайта <http://www.usabilitynet.org/home.htm>.

- фокус группы;
- мозговой штурм;
- оценка существующей системы;
- карточная сортировка;
- диаграммы сходства;
- сценарии (примеры);
- анализ задач;
- собрание по составлению требований.

На этом этапе проводится исследование пользователей и аспектов юзабилити, на основе чего в дальнейшем составляется спецификация требований будущей системы. Методы, используемые на данном этапе, предназначены для сбора информации о пользовательском интерфейсе, пользователях, их задачах и обстановке, в которой они выполняются. Собранные информация согласовывается и используется для составления технических требований.

Информацию для юзабилити-тестирования *собирают* с помощью опросов пользователей, бесед с ними, а также бесед в заданном контексте, наблюдений за пользователями и исследований на месте, а также с помощью анализа контекста использования, фокусных групп, либо с помощью мозгового штурма и оценки существующей системы.

Информацию затем *структурируют* с помощью методов карточной сортировки, диаграмм сходства и разработки сценариев (примеров).

Для *изучения* задач, которые выполняют пользователи в системе, прибегают к методу анализа задач.

Требования к создаваемой системе *согласуются* на собрании по составлению требований.

Касательно юзабилити технические требования:

- описывают весь диапазон первичных и вторичных пользователей системы;
- определяют приоритеты по каждому требованию в списке на основе согласованных критериев;
- определяют целевые показатели, по которым будет оцениваться юзабилити созданной системы;
- освещают юридические аспекты предъявляемых требований, ссылаются на признанные стандарты или рекомендации.

3. Этап дизайна и проектирования:

- руководства по дизайну;
- бумажные прототипы;
- эвристическая и экспертная оценка;
- параллельный дизайн;
- раскадровка;
- оценка прототипа;
- шаблоны.

4. Этап реализации и программирования:

- стилевые правила;
- быстрые прототипы.

5. Этап тестирования и оценки:

- диагностическая оценка;
- тестирование производительности;
- *Subjective Assessment (testing & post-release)*;
- эвристическая оценка;
- *Critical Incident Technique Analysis*;
- *Pleasure based approach*.

6. Этап выпуска:

- *Post release testing and measurement*;
- *Subjective Assessment (testing & post-release)*;
- опросы пользователей;
- *Remote evaluation*.

Список и краткое содержание наиболее часто применяемых методов юзабилити:

1. *Собрание участников проекта* — это стратегический метод, позволяющий вывести технические требования к продукту из целей, поставленных бизнесом, а также привлечь общее внимание к юзабилити-содержанию будущего проекта. Кроме того, собрание позволяет собрать информацию о назначении будущей системы и контексте ее использования.

2. *Анализ контекста использования*. С помощью метода анализа контекста (*context analysis*) собирается следующая информация:

— Кто будет пользователем системы и какие задачи он будет с её помощью выполнять? Почему он будет пользоваться системой? Каким опытом и навыками он обладает?

— Каковы технические характеристики системы и условия ее применения? На каком оборудовании будет работать система и пользователь? Каковы технические и природные условия работы?

Эта информация важна для составления списка требований к будущей системе и для планирования других методов. Этот метод может быть использован на первом этапе — этапе планирования и оценки, либо более интенсивно — на втором этапе: этапе составления требований.

3. *ISO 13407: Human centered design processes for interactive systems* — международный стандарт. Содержит рекомендации по организации процесса проектирования интерфейсов и встраиванию этого процесса в процедуру проектирования программных продуктов. В стандарте описаны методы юзабилити, необходимые для определения контекста использования продукта, выявления требований пользователей и заказчиков к системе, прототипирования и юзабилити-тестирования продукта.

4. *Usability Planning* — планирование юзабилити. Цель данного этапа — обеспечить эффективное использование методов в процессе дизайна и разработки, обеспечить постоянное влияние этих методов на проектировщиков с самой ранней стадии проекта.

5. *Анализ конкурентов* — позволяет перед началом работы над прототипами определить сильные и слабые стороны конкурента. На заседании участников проекта проводится демонстрация продуктов 4–10 наиболее популярных конкурентов, чтобы понять, как выполняются на них типичные задачи. Далее проводится обсуждение достоинств каждого из конкурентов, и в конце заседания составляется краткое предложение.

6. *Опрос пользователей (user survey)* — это один из способов узнать, как программой или веб-сайтом будут пользоваться конкретные группы пользователей, а также способ узнать, что собой представляют эти пользователи. Вопросы в опросных листах должны быть связаны с темами, которые важны для команды разработчиков. Обычно опросы проводятся по почте, но сейчас все чаще для этой цели используется Интернет.

7. *Метод бесед с пользователями (interview)* служит для выяснения фактов и мнений потенциальных пользователей о будущей системе. Обычно беседа проводится «один на один». Отчет о беседе необходимо тщательно проанализировать и выделить полезную информацию.

8. *Метод беседы в контексте (contextual inquiry)* — это специфический тип беседы с целью сбора информации от пользователей «в полевых условиях». Обычно беседа проводится «один на один» с пользователем. Цель беседы — получить как можно больше разнообразных данных от пользователей для последующего анализа.

9. *Наблюдение за пользователями исследования на месте (observational methods)* заключается в том, что разработчик наблюдает за тем, как пользователи работают, делая записи о том, что происходит на рабочем месте. Наблюдение может быть как прямым, когда разработчик присутствует непосредственно при выполнении пользователем задания, либо косвенным, когда выполнение задания просматривается на видеозаписи. Этот метод особенно полезен на раннем этапе при сборе данных для составления спецификации требований, а также при изучении того, как в настоящее время выполняется та или иная задача или процесс, под которую разрабатывается сайт или продукт.

10. *Фокус-группа (focus group)* — это неформальное собрание пользователей, у которых спрашивают мнение по определенной теме. Цель данных опросов — выяснить у пользователей их отношение и восприятие темы, а также их идеи и мнения по ней.

11. *Мозговой штурм (brain storming)* является самым старейшим методом разработки. Группа людей собирается вместе и усиленно думает над проблемой или предложением. Работа проводится в два этапа: на первом генерируются идеи, а на втором — анализируются. Очень важно, когда всем процессом руководит опытный специалист.

12. *Оценка существующей системы.* Оценка предыдущей версии сайта/программы (*existing system evaluation*) или оценка аналогичной системы конкурентов позволяет выявить проблемы с юзабилити и найти отправную точку для улучшения.

13. *Метод карточной сортировки (card sorting)* используется для нахождения скрытой структуры в нерассортированном списке предметов, понятий или идей. Исследователь пишет каждое понятие или предложение на маленькой карточке и затем просит каждого из шестерых (или более) пользователей разложить эти карточки по группам или кучкам. Результаты каждой сортировки затем комбинируют и, если необходимо, анализируют статистически.

14. *Диаграммы сходства (affinity diagramming)* используют для сортировки больших объёмов данных по логическим группам. Имеющиеся и/или новые понятия записывают на клейщихся карточках и затем разбивают по категориям группой пользователей.

Диаграммы сходства также могут использоваться

- для анализа данных, полученных при исследованиях на местах;
- для определения и группировки функций, которые будут присутствовать в продукте или на сайте;
- для анализа данных, полученных в результате тестов и оценок юзабилити.

15. *Сценарии (scenarios)* детально описывают то, как пользователи решают конкретные задачи в определенном контексте. Сценарии используют как примеры поведения пользователей, которые учитывают при разработке системы, а потом берут за основу при проведении тестов на юзабилити. Сценарии — это примеры (*use cases*) с акцентом на поведении пользователя и на решаемых им задачах.

16. С помощью *анализа задач (task analysis)* определяют, что пользователю нужно сделать для выполнения той или иной задачи. Детальный анализ задач используется для понимания существующей системы и того, как в ней движется информация. Эти знания важны как для поддержания существующей системы, так и для правильного построения новой. С помощью этого метода можно точно определить, какие функции должны быть включены в систему и как должен выглядеть интерфейс, обеспечивающий реализацию этих функций.

17. *Собрание пользователей и разработчиков (requirements meeting).* На нём определяются требования к юзабилити системы, которые затем будут проверяться в процессе разработки.

18. *Руководства по дизайну (design guidelines)* пользовательского интерфейса содержат набор готовых отработанных решений и дают как общие, так и подробные наставления по дизайну. Следование определенному руководству по дизайну может быть записано в требованиях к проекту. Дизайнеры и разработчики должны затем ознакомиться с указанными руководствами, и эти же руководства будут использоваться при экспертной оценке дизайна.

19. *Бумажные или какие-либо другие прототипы (paper and machine prototypes)* используются для того, чтобы уточнить требования к дизайну и получить первые варианты дизайна, которые могут быть быстро созданы и протестированы.

20. *Эвристическая и экспертная оценка.* Эвристическая оценка (*heuristic evaluation*) — это форма исследования юзабилити, в которой специалисты по юзабилити оценивают каждый элемент интерфейса, опираясь на список определенных принципов юзабилити (которые называются «эвристиками»). Экспертная же оценка (*expert evaluation*) —

это то же самое, только без опоры на специфические эвристики. Как правило, в оценке системы принимают участие два — три аналитика, которые проверяют дизайн системы на соответствие оговоренным правилам и принципам. Они записывают свои наблюдения и сортируют их в порядке серьезности обнаруженных проблем. В качестве аналитиков приглашают специалистов в области взаимодействия человека и компьютера или специалистов по человеческому фактору, но, как показывает опыт, и менее опытные специалисты способны обнаружить серьезные проблемы. Эвристическую или экспертную оценки можно проводить на различных этапах разработки, хотя лучше всего, если уже в какой-либо форме проведено исследование *контекста использования системы*. Это позволит специалистам сконцентрировать свое внимание на обстоятельствах использования создаваемой системы.

21. *Параллельный дизайн (parallel design)* — это метод, при котором создаются несколько альтернативных дизайнов (например, интерфейса) одновременно двумя-четырьмя группами разработчиков. Цель данного метода — оценить различные идеи до того, как будет принята единая концепция для реализации. Группы разработчиков должны работать независимо друг от друга, так как цель метода — создать как можно больше различных решений. Группы разработчиков не должны обсуждать друг с другом свои работы до тех пор, пока окончательные наброски не будут готовы для общего собрания. Задуманная система может опираться на один из этих дизайнов или их сочетание, в котором учтены лучшие идеи из каждого варианта.

22. «*Раскадровка (storyboard)*» — это грубый прототип, состоящий из серии набросков экранов/страниц. Они используются дизайнерами для того, чтобы проиллюстрировать или свести воедино свои идеи и получить отзывы по ним. Они особенно часто используются при создании мультимедийных презентаций.

23. *Оценка прототипа*. Оценка бумажного или машинного прототипа (*paper or machine prototype evaluation*) при участии пользователя имеет целью обнаружить проблемы с юзабилити. При проведении теста пользователя просят объяснить возникшие проблемы и пояснить, какой реакции он ожидал от системы на свои действия.

24. *Метод под названием «Гудвин Великий и Ужасный» (The Wizard of Oz)* позволяет произвести оценку технологии, которая еще не реализована в коде. Делается это с помощью человека, который имитирует ответную реакцию системы.

25. *Шаблоны для дизайна интерфейса (interface design patterns)* представляют собой готовые решения часто встречающихся проблем и ситуаций в процессе дизайна интерфейса. Конечные пользователи и

разработчики рассматривают интерфейсы с точки зрения того, какое шаблонное решение было использовано при его создании. Эта методология разработки интерфейсов еще пока не отработана, а примеры пока не изучены, однако это — многообещающий подход, который поддерживается большим числом сайтов, предлагающих информацию о стандартных решениях.

26. *Стилевые правила (style guides)* служат для того, чтобы создать единообразие интерфейса и его поведения. Стилиевые правила утверждаются как часть требований к системе, а соблюдение правил должно постоянно подвергаться проверке на протяжении всего этапа разработки.

27. *Быстрые прототипы* — используются специальные программные средства для быстрого создания прототипов.

28. *Диагностическая оценка*. В случае, когда качество работы системы признано неудовлетворительным, часто необходимо провести исследование вклада различных элементов контекста использования, а также взаимодействий между ними для определения главных причин проблемы. Этот процесс может также использоваться для определения элементов, которые поддаются изменению, для того, чтобы произвести повышение качества рабочей системы в использовании. Действия по диагностике, касающиеся контекста использования, — часто необходимый шаг в задании и оценке юзабилити продукта, поскольку они дают основания судить о том, может ли проблема быть решена перепроектированием продукта.

29. *Тестирование производительности* — оценка производительности. Используются методы оценки, связанные непосредственно с выполняемой деятельностью, и показатели, оценивающие время выполнения отдельных операций. Оценка производительности взаимодействия пользователей с системой обычно заключается в том, что группу пользователей просят выполнить определенный набор заданий; при этом замеряется время, которое им для этого потребовалось, число ошибок, которое они совершили, а также при необходимости некоторые другие параметры (частота обращения к справочным материалам, число посещенных за время выполнения задания страниц на сайте и т. п.).

30. *Субъективная оценка пострелиза (testing & post-release)* — субъективная оценка в процессе самого тестирования и тестирования пострелиза продукта.

31. *Анализ критических инцидентов (Critical Incident Technique Analysis)* — техника анализа критических инцидентов.

32. *Оценка удовольствия работы с продуктом (Pleasure based approach)* — приложения, оценивающие удовольствие от работы с продуктом.

33. *Измерение и тестирование пострелиза (Post release testing and measurement)* — тестирование и измерение параметров пострелиза продукта.

34. *Удаленная оценка (Remote evaluation)*. Главная идея метода удаленной оценки состоит в том, что должен проводиться интеллектуальный анализ прикладных регистраций (зарегистрированных в течение тестирования пользователя) с использованием информации модели задачи, для сравнения фактического поведения пользователя, содержащегося в файлах системного журнала, с запланированным пользовательским поведением, заключенным в модели задачи. Такая автоматическая оценка также обеспечивает оценщика набором мер относительно группы пользователей и позволяет идентифицировать проблемы применимости, полученные из недостатка соответствия между тем, как пользователи исполняют задачи, и моделью задачи системы.

3.3. Процесс исследования и его участники

Крайне важно, чтобы выборка участников тестирования была репрезентативной. Если число испытуемых невелико, то нужно стараться найти таких, кто больше всего подходит под описание «среднестатистического пользователя системы». Если же предполагается задействовать достаточно много людей, то нужно стремиться подобрать представителей различных подгрупп, которые можно выделить среди всех пользователей продукта (например, опытные пользователи и новички, представители различных возрастных групп и т. д.).

Основная идея юзабилити-тестирования заключается в том, чтобы произвести наблюдение за тем, как реальные пользователи взаимодействуют с программным продуктом в условиях, максимально приближенных к реальным. Для этого организаторам необходимо правильно подобрать задания (иногда их еще называют «тестовыми сценариями»), которые будут предложены для выполнения участникам тестирования. Так же, как и подбор участников, подбор заданий должен быть репрезентативным. Они должны соответствовать тем задачам, которые пользователи регулярно решают с помощью изучаемой системы в реальной жизни. Задания не следует делать чересчур объемными, но в то же время они должны охватывать большую часть функций системы.

Единого мнения на счет выбора оптимального числа участников тестирования нет. Р. Вирзи, Я. Нильсен и Т. Ландауэр еще в начале 90-х годов пришли к выводу, что пяти человек вполне достаточно, чтобы

обеспечить достаточно хороший результат при наилучшем соотношении затрат к количеству обнаруженных проблем. По настоящее время это мнение является преобладающим среди юзабилити-специалистов. Однако имеются сомнения в его справедливости, особенно при оценке крупных интернет-ресурсов. Есть мнения, что этого числа на самом деле недостаточно для сколько-нибудь качественной оценки больших сайтов.

Например, В. Головач считает, что для эффективной оценки крупного корпоративного сайта необходима выборка, состоящая не менее чем из двадцати человек. Но, учитывая, что первые несколько респондентов все-таки выявляют большое количество самых серьезных недостатков, он предлагает использовать тестирование сериями по несколько человек, после каждой из которых обнаруженные проблемы устраняют. Тестирование может продолжаться до тех пор, пока число выявляемых проблем не упадет до достаточно низкого уровня.

У. Хадсон отмечает, что тестирование с пятью пользователями дает хорошие результаты, когда они работают с четко определенными тестовыми сценариями, которые однозначно указывают на то, на каких элементах сайта должен сконцентрироваться испытуемый. Если же задания дают в более свободной форме, то исследования показывают, что должной эффективности добиться обычно не удается. Однако в случае с крупными сайтами и другими сложными объектами возникает вопрос: можно ли с помощью жестко определенных заданий в достаточно полной степени воспроизвести процесс взаимодействия с ними в реальной жизни?

Нильсен призывает больше внимания уделять итеративному тестированию. Пусть даже пять респондентов не всегда смогут сразу выявить достаточное число проблем, но повторное тестирование с тем же числом участников после устранения выявленных недостатков окажется, по его мнению, более эффективным, чем изначальное привлечение десяти респондентов.

Что касается технического обеспечения процедуры тестирования, то наилучшим будет решение проводить его в специально оборудованной лаборатории, где имеется все необходимое оборудование, а участники будут чувствовать себя максимально комфортно. Для минимизации влияния на респондентов организаторы могут наблюдать за процессом из другого помещения, воспользовавшись соответствующей аппаратурой. Все происходящее желательно записывать на видео, чтобы впоследствии иметь возможность дать более подробную оценку полученным данным. При необходимости исследование можно проводить и в «полевых» условиях.

Перед началом тестирования организаторам необходимо четко определиться с тем, какие конкретно цели они хотят достигнуть с его помощью. От этого зависят формулировки тестовых сценариев и

конкретные методы тестирования. Например, если необходимо определить сильные и слабые стороны интерфейса и улучшить его, то стандартным подходом является метод «мысли вслух». Если же стоит задача выбора между несколькими различными интерфейсами, то для этого больше подходит оценка производительности и т. д.

3.4. Исследование пользователей

Исследование пользователей — важный этап разработки продукта. Он позволяет на начальном этапе разработки продукта дать ответ на вопросы: кто ваши *пользователи*? Зачем нужен разрабатываемый продукт, и как он будет использоваться? Применяются методы, оценивающие рыночное поведение пользователя, и контекст, в котором происходит принятие решения о покупке продукта.

3.4.1. Маркетинговые исследования

Подобно опросам и фокус-группам маркетинговые исследования могут стать ценным источником общей информации о пользователях. Эти методы особенно эффективны, если существует четкая формулировка того, какую информацию мы хотим получить с их помощью. Нужно ли нам знать, как ведут себя пользователи, когда взаимодействуют с конкретной функцией продукта? Или мы это уже знаем, но хотим выяснить, почему они ведут себя именно так? Чем четче вы опишите свои интересы, тем более точно и эффективно будут сформулированы вопросы, направленные на получение нужной информации.

3.4.2. Исследование контекста

Контекстуальное исследование — это набор методов по изучению контекста, возникающего в повседневной жизни пользователя. В основе этого инструментария лежат методы, применяемые антропологами при изучении культур и сообществ. Например, методы исследования поведения кочевых племен, примененные в более узком масштабе, позволяют изучить поведение людей, покупающих запчасти для самолетов.

Недостатками контекстуального исследования являются высокая стоимость и большие затраты времени, однако при наличии достаточных ресурсов и требований глубокого знания пользовательской аудитории контекстуальное исследование может открыть такие тонкости пользовательского поведения, о которых невозможно узнать с помощью других методов.

Анализ задач тесно связан с контекстуальным тестированием. В его основе лежит идея о том, что любое взаимодействие пользователя с продуктом (сайтом) происходит в контексте некоторой задачи, решаемой

пользователем. Иногда это — очень узкая задача (например, покупка билетов в кино), иногда — широкая (например, выяснение правил международной торговли).

Анализ задач является методом подробного изучения шагов, предпринимаемых пользователями при решении своих задач. Это изучение проводится либо с помощью интервью, в ходе которых мы просим пользователей поделиться опытом, либо с помощью непосредственных наблюдений за пользователями в естественной среде.

Контекстуальный опрос производится тем же порядком, что и опросы при полевых исследованиях или интервьюирование. Тем не менее он обладает рядом особенностей. Например, интервьюирование в процессе исследования методом контекстуального опроса, как правило, не включает заранее определенных вопросов, записанных в подробных формулировках. Вместо этого благодаря партнерству интервьюируемого и интервьюера возникает диалог, в котором интервьюер сможет не только узнать о мнении и опыте интервьюируемого, но и понять его мотивации, и уяснить контекст.

Одно только присутствие экспериментаторов может сделать испытуемого более раздражительным и агрессивным. Выступая в роли берущего интервью, вам необходимо войти в мир интервьюируемого, стать частью его. Иногда проходит значительное время, прежде чем интервьюируемые испытуемые привыкнут к вашему присутствию. С этого момента работа становится более легкой, поскольку пользователи, с которыми возник контакт, рассказывают, что они *в действительности* думают о вашем продукте.

Важно выяснить, с кем именно проводить интервью. Нужно выбрать из команды нужных вам специалистов путем специального наблюдения. Желательно подбирать лиц уравновешенных, с развитыми речевыми навыками, по возможности имеющих образование. День наблюдений за такими специалистами даст вам достаточно информации для оценки качества проекта.

3.5. Контрольные вопросы

1. В чем специфика и основные черты юзабилити-тестирования как экспериментального метода?
2. Кто такие «модераторы», «респонденты» и «фасилитаторы»?
3. В чем смысл юзабилити-тестирования как экспериментальной процедуры?
4. С какой целью модератор и респондент юзабилити-тестирования должны находиться в разных помещениях?

5. Дайте определение юзабилити программного обеспечения.
6. Назовите пути оценки юзабилити продукта.
7. Как оценить юзабилити путем анализа процесса взаимодействия между пользователем и продуктом?
8. Опишите этапы проектирования программных интерфейсов и методы юзабилити, используемые на этих этапах.
9. Какие методы юзабилити используются на этапе составления требований к проекту?
10. Какие методы юзабилити используются на этапе тестирования и оценки программного продукта?
11. Кратко опишите анализ контекста использования продукта.
12. Как осуществляется наблюдение за пользователями в юзабилити-тестировании?
13. Что такое фокус-группа?
14. Опишите особенности применения метода «мозгового штурма» и его основные этапы.
15. Опишите метод сценариев.
16. Чем различаются между собой методы эвристической и экспертной оценки?
17. Сформулируйте требования к проведению метода параллельного дизайна.
18. Что такое «метод шаблонов»?
19. Опишите методы формирования репрезентативной выборки для проведения юзабилити-экспериментов.
20. С какой целью проводят исследование пользователей?
21. Что такое «исследование контекста», и какими методами оно производится?

3.6. Темы для дискуссии

1. Перспективы использования новых физических принципов в задачах юзабилити-тестирования.
2. Опишите идеальную лабораторию юзабилити.
3. Факторы, влияющие на эффективность юзабилити-тестирования.

Глава 4. Основные методы юзабилити-тестирования программных интерфейсов

4.1. Анализ рабочих заданий

Анализ рабочих заданий (task analysis) — набор методик, использующих анкетирование или открытое интервью для формирования детального представления о том, как люди в настоящий момент выполняют конкретные задания. Это — один из профессиографических методов, широко используемых в инженерной психологии и эргономике для описания деятельности пользователя.

В таком исследовании наиболее интересны следующие вопросы:

- стоящая за заданием реальная цель — ответ на вопрос о том, для чего пользователь выполняет задания;
- частота и важность выполнения задания;
- триггеры — что служит поводом или сигналом для выполнения задания;
- зависимости — что требуется для выполнения задания, и что зависит от ее выполнения;
- люди, которые вовлечены в выполнение задания, их роли и зоны ответственности;
- конкретные действия, которые требуется выполнить;
- решения, которые необходимо принять;
- информация, которая нужна для принятия решений;
- ошибки и исключительные ситуации — что может пойти не так;
- способы исправления ошибок и обработки исключений.

После того как заполнены анкеты или проведены интервью, выполняется формальная декомпозиция и анализ рабочих заданий. Как правило, это делается при помощи диаграммы потоков или сходной диаграммы, передающей отношение между действиями и отношения между людьми и процессами.

Пользовательские исследования — фундамент проектирования. Необходимо уделять планированию исследований достаточное время. Следует подбирать методы исследования сообразно этапу разработки, что пойдет на пользу разрабатываемому продукту и поможет избежать потерь времени и ресурсов. Лабораторное тестирование продукта может дать

большой объем информации, но не обязательно будет иметь большую ценность. Когда концепция построена на основе качественных пользовательских исследований и моделей, опирающихся на результаты этих исследований, юзабилити-тестирование становится еще более эффективным инструментом для оценки решений, принятых в ходе проектирования.

4.2. Сегментация пользовательской аудитории

Пользовательские потребности можно разбить на части путем сегментирования пользовательской аудитории. Деление на различные группы производится с помощью выделения сходных ключевых характеристик представителей каждой группы. Существует большое множество различных способов сегментирования целевой аудитории. Выбор способа зависит от целевой аудитории, особенностей разрабатываемого продукта и т. п.

На первых этапах сегментирования может получаться большое количество групп, но в процессе определения приоритетов некоторые группы объединяются. В результате получаются 3–4 основные целевые группы.

Для того чтобы лучше осознать, какие цели преследуют пользователи, необходимо сначала понять, кто они. Чтобы сделать пользователей более конкретными и понятными, следует описывать (персонифицировать) типичных представителей каждой группы целевой аудитории, формируя репрезентативную выборку. Это должна быть выборка, отражающая свойства генеральной совокупности, в которую входят все возможные пользователи, на которых направлено внимание разработчиков продукта.

4.3. Персонажи

Персонажи — модели пользователей, создаваемые с целью одушевления процесса разработки. Придавая облик и имя разрозненным элементам данных, полученным в результате исследования и сегментации пользовательской аудитории, персонажи помогают разработчикам постоянно помнить о пользователях в ходе работы над проектом.

Исследование потребностей пользователей является своего рода маркетинговым исследованием, целью которого является выявление потребности потребителя, что в итоге позволит создать наиболее подходящий продукт. Исследование пользовательских потребностей позволяет более объективно представить себе результат разработки.

Часто аудитория потенциальных пользователей представляет собой довольно пеструю публику с различными целями и потребностями. Для наиболее четкого представления основных приоритетов в дальнейшем

процессе разработки целесообразно использование процедуры сегментации аудитории потенциальных пользователей по группам.

Каждая отдельная группа пользователей преследует свои интересы и стремится к достижению целей, которые могут быть совершенно не похожими на цели участников других групп.

4.4. Оценка производительности

Оценка производительности взаимодействия пользователей с системой (*Performance measurement*) заключается в том, что группу пользователей просят выполнить определенный набор заданий. При этом измеряют время, которое им для этого потребовалось, число ошибок, которое они совершили, а также при необходимости некоторые другие параметры (частота обращения к справочным материалам, число посещенных страниц на сайте и т. п.).

При проведении такой оценки общение между организатором и респондентом должно сводиться лишь к объяснению заданий, которые последнему необходимо выполнить. В остальное время организатор должен играть роль пассивного наблюдателя и не вмешиваться в процесс работы пользователя, если только тот не оказался в ситуации, когда он не способен самостоятельно разрешить возникшую проблему. Только при соблюдении этого требования можно рассчитывать на объективность получаемых данных.

Важной частью такого подхода является анализ полученной информации. Необходимо уметь правильно интерпретировать собранные данные. Количественные данные вряд ли могут показать, в чем конкретно заключается проблема, зато они хорошо подходят для определения того, где сосредоточены наиболее серьезные недостатки, или для сравнения различных интерфейсов. Да и сами результаты измерений могут быть обусловлены различными факторами. Возможно, что ошибка вызвана не серьезными проблемами в системе, а просто невнимательностью конкретного пользователя. Тем не менее все ошибки необходимо рассматривать более подробно, стремясь определить вызвавшие их причины.

Помимо характеристик, которые можно объективно измерить, существует такой параметр, как удовлетворенность пользователя от работы с системой. Его также желательно учитывать. Для этого с респондентом нужно провести небольшую беседу или попросить его заполнить специальную анкету после завершения выполнения заданий.

Другой вариант действий — наблюдать за поведением пользователя во время работы с системой и попытаться понять ощущения от взаимодействия по косвенным признакам и реакциям.

4.5. Мысли вслух

Этот метод Я. Нильсен считает одним из самых важных в арсенале юзабилити-специалиста. Он достаточно прост и заключается в том, что во время работы с тестируемой системой пользователь непрерывно проговаривает вслух все возникающие у него мысли. За счет этого юзабилити-специалист может понять, как именно пользователь воспринимает представленную систему и чем обусловлены те или иные его действия. С помощью такого подхода даже от небольшого числа респондентов можно получить большое число сведений о существующих проблемах и возможных улучшениях. Наблюдатель может подталкивать пользователя наводящими вопросами, но при этом не должен давать ему никаких подсказок касательно интерфейса системы.

Слабая сторона этого подхода заключается в том, что он не слишком хорошо сочетается с методами, которые ориентированы на количественную оценку производительности взаимодействия пользователя с программным продуктом. Это связано с тем, что необходимость проговаривать все свои действия снижает скорость работы. Кроме того, требование озвучивать свои мысли для многих людей может быть весьма непривычным, и они могут испытывать дополнительные затруднения при тестировании. Наконец, сама вербализация мыслей может оказывать влияние на поведение пользователей, часто заставляя их действовать более собранно и логично. Существуют исследования, которые показывают, что при размышлениях вслух пользователи несколько лучше справляются с поставленными задачами.

4.6. Конструктивное взаимодействие

Это — разновидность метода «мысли вслух», которая включает совместную работу сразу двух участвующих в эксперименте пользователей с системой. Основное его достоинство в том, что общаясь между собой пользователи оказываются в более естественной ситуации, чем когда их просто просят проговаривать свои действия поодиночке. Потенциальной проблемой для обеспечения конструктивного взаимодействия может оказаться несовместимость двух респондентов и различие в их подходах к работе с интерфейсом. Методика лучше всего подходит для ситуаций, когда есть возможность привлечь к тестированию большое количество участников, что проявляется при создании массовых продуктов.

4.7. Фокус-группы

Метод фокус-групп (*Focus groups*) имеет давнюю историю и применяется в различных маркетинговых исследованиях еще с 50-х годов XX века. Я. Нильсен достаточно подробно рассмотрел методику фокус-

групп и особенности ее применения в юзабилити. Он характеризует фокус-группы следующим образом: *«достаточно неформальная техника, которая может помочь определить потребности и ощущения пользователей как до начала проектирования интерфейса, так и уже после его реализации»*. Фокус-группа — это метод группового интервью.

Цель фокус-группы заключается в сборе качественных данных на основе подробного обсуждения участниками конкретной проблемы (вопроса). Метод фокус-групп позволяет понять, что на самом деле люди думают и чувствуют при работе с проектируемым продуктом. В работе каждой фокус-группы принимают участие от 4 до 12 человек. Это — люди, предварительно отобранные в соответствии с определенными критериями. Обсуждение проблемы (вопроса) в фокус-группе проходит в условиях, близких к естественным, возникающим при работе с продуктом.

Групповое обсуждение (рис. 2) фокусируется на определенной тематике. Вопросы к участникам и план проведения фокус-группы тщательно готовят заранее. Групповую дискуссию организует профессиональный интервьюер. Он выполняет функции модератора. Его задача — создать в группе безопасную и комфортную атмосферу, побудить участников свободно высказываться по заданной теме и выслушивать мнение всех.

В среднем участие в фокус-группе длится от 1 до 2 часов. После завершения дискуссии результаты обсуждения подвергают тщательному анализу, проводят обобщение и систематизацию собранной информации.



Рис. 2. Обсуждение в фокус-группе

4.7.1. Процесс исследования и его участники

Метод фокус-групп заключается в том, что несколько пользователей собираются в группу и обсуждают вопросы, так или иначе касающиеся рассматриваемого продукта. Обсуждение обычно продолжается около 2 часов и регулируется организатором — «ведущим».

Применение фокус-групп — это сравнительно затратное решение, так как в них минимальное число участников больше, чем в других исследованиях. Кроме того, для обеспечения требуемой точности результатов необходимо проводить несколько сеансов с различными респондентами.

4.7.2. Особенности методик

Многие исследователи предостерегают от чрезмерного увлечения методом фокус-групп при решении задач юзабилити. Это часто происходит, когда в разработке веб-сайтов и других программных продуктов принимают участие люди, связанные с маркетингом и рекламой, откуда изначально и происходит эта методика.

В данном случае фокус-группы следует использовать прежде всего для того, чтобы определить, какие функции пользователи хотели бы видеть в конечном продукте, в то время как вопросы, связанные с организацией интерфейса и особенностями его восприятия пользователями, должны рассматриваться с применением других методик. Дело в том, что в отличие от большинства других объектов, для исследования отношения к которым применяются фокус-группы, программные продукты должны не просто нравиться пользователям, а необходимо, чтобы пользователи были еще и в состоянии эффективно ими пользоваться. Единственным надежным способом проверить это является наблюдение за индивидуальным взаимодействием пользователя с рассматриваемой системой. Фокус-группы чаще всего не подразумевают такой возможности: обычно участникам просто демонстрируют, как с ней работает кто-то другой. При этом впечатления от наблюдения за чужими действиями сильно отличаются от впечатлений, которые человек получает от самостоятельного общения с продуктом.

Просмотр демо-версии практически не порождает такие вопросы, как «что делать дальше?» или «зачем нужна эта кнопка?». Даже если участникам предлагается самостоятельно поработать с системой, а после чего в группе обсудить возникшие у них затруднения (своеобразный вариант юзабилити-тестирования, с помощью которого организаторы пытаются сократить временные затраты), то полученные данные все равно будут гораздо беднее, чем при наблюдении за индивидуальной работой.

Пользователи зачастую не способны достаточно точно описать свои ощущения от взаимодействия с системой, тем более в присутствии других людей. Как и другие схемы, базирующиеся на опросе пользователей, а не на наблюдении за ними, применение фокус-групп может давать неточные результаты. Ведь пользователи далеко не всегда хорошо представляют, что им нужно на самом деле. Чтобы минимизировать эту проблему, Нильсен предлагает по возможности максимально сужать обсуждаемые вопросы,

как можно больше конкретизируя их. Например, спрашивать мнение участников не о какой-то идее или технологии вообще, а о конкретных примерах ее применения.

Предпочтительно проведение нескольких сеансов с различными участниками, так как существует вероятность ухода дискуссии в сторону от обсуждаемых вопросов и не репрезентативности результатов.

Здесь играет важную роль фигура ведущего, который должен следить за тем, как проходит обсуждение, и не давать ему отклоняться от интересующих исследователей аспектов, а также не допускать того, чтобы мнение одного из участников оказывалось доминирующим. С другой стороны, ведущему необходимо соблюдать баланс и не загонять дискуссию в чересчур тесные рамки, лишая пользователей возможности свободно высказывать свои идеи.

Исходя из вышеуказанных проблем, можно сказать, что в исследованиях, связанных с юзабилити-характеристиками программных продуктов, фокус-группы имеет смысл применять лишь на начальных этапах проектирования, чтобы выявить основные потребности и ожидания потенциальных пользователей. Применение их для оценки существующих или проектируемых интерфейсов не оправдано и не дает положительных результатов.

4.8. Метод карточной сортировки

Карточная сортировка (*card sorting*) является быстрым, недорогим и надежным методом, который помогает продумать общую структуру и принципы размещения информации на сайте, а также некоторые вопросы, касающиеся навигации, организации меню и т. п.

Карточная сортировка — техника, снижавшая популярность благодаря информационным архитекторам. Она позволяет понять, как пользователи организуют идеи и информацию. Существует ряд вариантов этой техники, но обычно она сводится к тому, что пользователи выполняют сортировку колоды карт, каждая из которых описывает определенную функциональность продукта или веб-сайта либо содержит связанный с ним фрагмент информации. Сложной частью данной техники является анализ результатов: необходимо найти паттерны и зависимости путем выявления тенденций или посредством статистического анализа.

Карточная сортировка, несомненно, является ценным инструментом для раскрытия определенных аспектов пользовательской *ментальной модели*, однако эта техника предполагает, что респондент обладает хорошими навыками организации информации и то, как он сортирует набор абстрактных тем, напрямую связано с подходом, который он, в конце концов, выберет, когда пожелает воспользоваться вашим продуктом.

Так получается не всегда. Один из способов преодолеть возможное расхождение — попросить пользователя упорядочивать карты, подготовленные на основе способов выполнения задач, решать которые призван проектируемый продукт. Другой способ повысить ценность карточных исследований — после завершения процедуры побеседовать с респондентами, чтобы выявить принципы организации информации, которые они применяли при сортировке.

4.8.1. Процесс исследования и его участники

Метод заключается в том, что приглашенные пользователи получают особый набор карточек, на каждой из которых написан определенный элемент содержимого сайта или его функционального наполнения. Они должны быть перемешаны. Затем участники должны сгруппировать эти карточки таким образом, который покажется им наиболее логичным и очевидным (рис. 3). Результат этого процесса поможет понять, как респонденты структурируют информацию у себя в голове, и в какой форме они ожидают увидеть ее на сайте. Это важно, так как взгляд разработчика может серьезно отличаться от взгляда обычного пользователя.

Чтобы обеспечить высокую эффективность от применения этой методики, предварительно следует провести тщательный анализ того, какое содержимое и какие функции пользователи хотели бы видеть на соответствующем сайте (например, с помощью фокус-групп, опросов, анализа статистической информации).



Рис. 3. Сортировка карточек в методе

Список сортируемых элементов не должен оказаться перегружен, иначе их группировка займет слишком много времени, а качество результатов снизится. В то же время чересчур малое число карточек может сделать невозможным их осмысленное распределение по категориям. Д. Спенсер и Т. Уорфел в статье «*Card sorting: a definitive guide*»

высказывают мнение, что оптимальное число карточек находится в диапазоне от 30 до 100. Они также отмечают, что следует давать участникам несколько пустых карточек, чтобы те могли добавить кажущиеся им важными элементы. Карточная сортировка может выполняться как индивидуально каждым участником, так и в группах.

Д. Спенсер и Т. Уорфел отмечают, что групповое тестирование дает более ценные результаты, чем индивидуальное, так как несколько человек, во-первых, в состоянии качественно обработать большее количество элементов, а во-вторых, постоянно ведут обсуждение между собой, что позволяет наблюдателю лучше понять механизмы мышления пользователей. Однако индивидуальную сортировку обычно проще организовать, и она может быть актуальна при работе над не слишком крупными проектами.

Я. Нильсен считает, что метод карточной сортировки требует участия большего количества людей, чем многие другие применяемые для оценки юзабилити методы, и оптимальное число участников такого рода исследований составляет 15 человек. Это число позволяет добиться достаточно хороших результатов и дает корреляции 90 % с «абсолютной» оценкой (ситуацией, когда в исследовании принимает участие значительно больше респондентов), чего, по мнению Нильсена, вполне достаточно. При этом дальнейшее увеличение количества испытуемых крайне медленно повышает точность исследования. Так, участие 30 человек обеспечивает лишь корреляции 95 %, при том, что затраты возрастают вдвое.

Д. Спенсер и Т. Уорфел также называют 15 участников оптимальным числом для групповой сортировки (их предлагается разбить на группы по 3 человека), а для индивидуальной предлагают использовать от 7 до 10 человек. Как и при других исследованиях, в случае применения карточной сортировки следует внимательно следить за тем, чтобы выборка участников была репрезентативной.

4.8.2. Особенности методики карточной сортировки

Существуют два основных способа проведения таких исследований: их называют «открытой» и «закрытой» карточной сортировкой.

При использовании метода *закрытой* карточной сортировки участники просят распределить полученные карточки по заранее predetermined категориям. *Открытая* же сортировка подразумевает, что они сами выделяют и описывают те группы, на которые, как им кажется, следует разбить полученные элементы. Считается, что открытая сортировка лучше подходит для проектирования новых сайтов или при тотальной смене дизайна. *Закрытую* карточную сортировку предпочтительно применять для внесения дополнений в уже

существующую структуру или для уточнения результатов, предварительно полученных с помощью открытой карточной сортировки. Метод карточной сортировки не предназначен для оценки существующих интерфейсов.

4.9. Экспертная оценка

В отличие от методик, что были описаны ранее, участниками экспертной оценки являются не предполагаемые пользователи, а эксперты в области юзабилити и профессиональные дизайнеры интерфейсов. Они исследуют продукт и стараются выявить имеющиеся, на их взгляд, проблемы. Экспертная оценка зачастую не может заменить собой полноценное юзабилити-тестирование, но у нее есть свои преимущества. Существуют различные формальные способы проведения такого рода оценок. Нильсен выделяет восемь различных вариаций (например, мысленное перемещение по интерфейсу, когда эксперт продумывает все возможные пути движения пользователя и ищет потенциально опасные места). Вряд ли нужно рассматривать все из них, остановимся лишь на методе эвристической оценки, который получил наибольшую популярность и распространение.

4.10. Эвристическая оценка

Основной особенностью данной методики (*Heuristic Evaluation*) является наличие списка определенных юзабилити-принципов (или эвристик). Участвующие специалисты поочередно исследуют продукт, выделяют существующие, на их взгляд, недостатки и классифицируют эти недостатки как нарушение одного или нескольких из этих принципов (рис. 4).



Рис. 4. Метод Heuristic Evaluation при анализе 3-степенного гаптического игрового контроллера

4.10.1. Процесс исследования и его участники

Список эвристик заранее определяется организаторами исследования. Они могут варьироваться, но наиболее известен набор из 10 правил, предложенный Я. Нильсеном. Они сформулированы как универсальные для всех программных продуктов:

1. *Видимость состояния системы.* Пользователь всегда должен быть информирован о том, что происходит в системе с помощью наглядных средств и в течение разумного времени.
2. *Соответствие между системой и аудиторией.* Система должна общаться с пользователем на привычном для него языке.
3. *Свобода действий пользователя.* Пользователи часто выбирают то или иное действие по ошибке, и у них должна быть возможность без лишних усилий вывести систему из нежелательного состояния.
4. *Последовательность и стандарты.* Пользователи не должны сомневаться, что означают те или иные обозначения или названия. Нужно следовать общепринятым в данной области стандартам и договоренностям.
5. *Предотвращение ошибок.* «Лучше заниматься не дизайном сообщений об ошибках, а дизайном, который бы предотвратил возникновение ошибок».
6. *Понимание лучше запоминания.* Все объекты, функции и действия в интерфейсе должны находиться перед глазами пользователя. Не нужно заставлять его держать в памяти ту информацию, что может быть запомнено системой. Инструкции по работе с конкретным элементом продукта всегда должны быть доступны при первой же необходимости.
7. *Гибкость и эффективность использования.* В продукте должны присутствовать функции, которые не очевидны для новичков и не мешают им, но позволяют повысить скорость и эффективность работы опытных пользователей. Потребности обеих групп должны быть удовлетворены.
8. *Эстетичный и минималистический дизайн.* Любая лишняя информация снижает уровень наглядности интерфейса и степень эффективности работы с ним.
9. *Определение, диагностирование и исправление ошибок.* Сообщения об ошибках должны быть понятны пользователю, четко определять причину возникших проблем и предлагать конкретный способ их решения.
10. *Справка и документация.* Вся справочная информация должна быть доступна, ориентирована на потребности пользователя, содержать конкретные шаги и иметь разумный объем.

Эвристическая оценка, которую дает каждый конкретный участник, во многом зависит от его личных характеристик и особенностей

восприятия, поэтому, хотя данную методику и можно использовать даже при наличии единственного эксперта, эффективность в таком случае будет значительно ниже, чем при групповой оценке. Один человек никогда не сможет обнаружить все имеющиеся в интерфейсе проблемы. Я. Нильсен отмечает, что пять экспертов зачастую в состоянии выявить около 75 % недостатков, и это — достаточно хороший результат. Три человека — это минимальное число участников, при котором имеет смысл проводить оценку.

4.10.2. Особенности методики

Так как в ходе эвристической оценки участники не используют систему как таковую (не выполняют те задачи, которые предположительно будут решать с ее помощью пользователи), то такую оценку можно применять и для оценки интерфейсов на самых ранних этапах разработки, когда они существуют лишь на бумаге. Методика подходит для выявления проблем юзабилити на различных стадиях — и для первоначального проектирования, и для доработки уже реализованных решений.

М. Леви и Ф. Конрад в своей статье «*Usability Testing of World Wide Web Sites*» пишут, что в контексте изучения веб-сайтов эвристическая оценка позволяет выявить лишь ограниченное число проблем, которые обычно носят локальный характер (не соответствующие содержанию заголовки и обозначения, не понятная для пользователя лексика, неоднозначные элементы дизайна и т. п.). При этом с ее помощью практически никогда не удастся обнаружить серьезные структурные проблемы, касающиеся организации всего сайта целиком. Они предлагают бороться с этой особенностью за счет дополнения эвристической оценки методом карточной сортировки, который как раз хорошо подходит для решения таких задач.

Похожее мнение высказывает и В. Головач, который считает, что «*эвристическая оценка не выявляет структурные проблемы интерфейса, которые, как правило, заметно важнее локальных проблем*».

Это частично соответствует наблюдениям Я. Нильсена, который считает, что с помощью данной методики удастся выявить больше мелких юзабилити-недостатков, чем крупных. Он, однако, отметил, что это свидетельствует лишь о том, что эвристическая оценка лучше, чем другие методики может помочь выявить небольшие проблемы. При этом серьезные недочеты от нее также не укрываются (следует учитывать, что Нильсен в данном случае говорит не о веб-сайтах, а о программных продуктах в целом).

4.11. Макетирование (прототипирование)

Начинать реализацию продукта, базируясь на непроверенных данных и предположениях, достаточно опасно: ведь потери ресурсов в случае необходимости исправления закрывшихся на стадии проектирования ошибок могут быть очень значительными, поэтому имеет смысл начинать юзабилити-тестирование как можно раньше. Для того чтобы вовлечь в процесс предполагаемых пользователей, можно использовать прототипы конечной системы, создание которых не потребует значительных затрат времени и средств (рис. 5).



Рис. 5. Структура метода прототипирования

Прототип неизбежно отличается от готового продукта по своим характеристикам. Выделяют два основных вида прототипов в зависимости от того, в чем заключается отличие. О вертикальном прототипе говорят, когда урезается число функций конечной системы, но те из них, что присутствуют, полностью работоспособны (в таком случае говорят о тестировании «в глубину»). В горизонтальном прототипе, напротив, обозначаются все предполагаемые функции продукта, но они не реализуются (тестируется главным образом дизайн интерфейса).

Согласно другому принципу разделения прототипов также можно выделить две группы — прототипы высокой точности (*high-fidelity*) и низкой (*low-fidelity*). Первые отличаются от вторых тем, что выглядят и ведут себя примерно так же, как и конечный продукт, и на первый взгляд

их трудно различить. А вот прототип низкой точности даже не слишком продвинутый пользователь ни за что не перепутает с настоящим интерфейсом.

В работе с прототипами существуют различные подходы. На рынке сейчас присутствует достаточно много программных продуктов, призванных упростить этот процесс. Все они имеют свои сильные и слабые стороны, и вряд ли есть смысл на них останавливаться. В качестве примера лучше более подробно рассмотреть метод, о котором Я. Нильсен отзывается следующим образом: *«с его помощью можно протестировать на пользователях самые ранние дизайнерские идеи с очень низкими затратами, что позволит исправить юзабилити-проблемы еще до того, как деньги будут потрачены на реализацию чего-то совершенно неработоспособного»*. Это — метод создания бумажных прототипов.

4.12. Бумажное прототипирование

Наиболее полным образом эта методика рассмотрена в книге Кэролин Снайдер *«Paper prototyping. The fast and easy way to design and refine user interfaces»*. Она дает следующее определение данному методу: *«бумажное прототипирование — это одна из разновидностей юзабилити-тестирования, во время которой пользователи выполняют реалистичные задачи, взаимодействуя с бумажной версией интерфейса, которой управляет человек, «играющий роль компьютера», не объясняющий, каким образом работает интерфейс»*.

Бумажные прототипы относятся к группе прототипов низкой точности.

4.12.1. Процесс исследования и его участники

К. Снайдер выделяет четыре роли, в которых выступают участники тестирования с применением бумажных прототипов. Это — пользователь, консультант (*facilitator*), «компьютер» и наблюдатель.

Тестирование проходит следующим образом — вначале организаторы определяют несколько типичных задач, которые предположительно будут решать пользователи разрабатываемого ресурса. Затем они делают скриншоты (если уже имеются какие-либо наработки и макеты) или просто нарисованные от руки наброски различных элементов интерфейса (различных страниц в случае с веб-сайтом). С ними и будет взаимодействовать пользователь. Его просят выполнить несколько задач, взаимодействуя непосредственно с бумажным прототипом.

Один или несколько человек из числа организаторов играют роль «компьютера», выкладывая перед пользователем листы бумаги таким образом, чтобы симулировать взаимодействие с настоящим интерфейсом,

но при этом не дают никаких пояснений о том, как он работает. Единственный, кто общается с пользователем, — это консультант, который объясняет ему суть эксперимента, дает задания и отвечает на возникающие вопросы. Остальные участники из числа организаторов наблюдают за происходящим со стороны. Наблюдатели могут находиться в помещении, где проводится тестирование, или следить за ним с помощью каких-либо технических средств.

4.12.2. Особенности методики

Прототипы вовсе не должны в подробностях передавать дизайн интерфейса (обычно это слишком трудоемко и не оправдывает усилий), только — раскладку основных элементов. Бумажное прототипирование эффективно выявляет проблемы, которые непосредственно не касаются графических особенностей интерфейса (содержимое, термины, структура и т. п.). К. Снайдер считает, что с помощью этого метода с большой долей вероятности можно обнаружить следующие виды проблем, связанные

- с терминологией и концепциями, которые оказываются неясны для конечного пользователя;
- с навигацией по сайту;
- с содержимым сайта (пользователь может не обнаружить того, что ожидал или не удовлетвориться тем способом, как оно представлено на сайте);
- со справочной информацией (в каких местах и какой форме пользователю требуется подсказка);
- с функциональностью (какие потребности пользователей были упущены при первоначальной оценке, а какие оказались невостребованными);
- с расположением различной информации на экране (каковы приоритеты пользователей, в каком порядке лучше воспринимаются те или иные данные).

А вот следующие недостатки вряд ли можно выявить, используя этот метод. Это — проблемы, связанные

- с длительным регулярным использованием продукта (то, что удобно для новичка, может вызывать раздражение у людей, знакомых с продуктом);
- с прокруткой веб-страниц (трудно отобразить на листе бумаге);
- с объемными документами и списками;

- с непреднамеренными ошибками пользователя (например, он знает, куда надо нажать, но случайно нажимает не туда);
- с размерами элементов (в прототипе размеры могут быть неточны);
- с выпадающими и каскадными меню;
- со временем реакции сайта на действия пользователя.

Это — лишь часть проблем, которые нельзя охватить с помощью метода тестирования бумажных прототипов, что нужно учитывать при проектировании интерфейсов.

Несмотря на то, что далеко не все типы проблем оказываются охваченными методом прототипирования, у этого метода имеются значительные преимущества. Основными достоинствами бумажных прототипов являются крайне низкие затраты ресурсов на их создание при достаточно высоком уровне эффективности. Они не требуют особых технических навыков, с их помощью можно получить важную информацию от пользователей на самой ранней стадии разработки до того, как будут потрачены усилия на реализацию того или иного подхода. Кроме того, их простота и дешевизна дают возможность испытать много различных вариантов вместо одного.

Интересно, что, несмотря на описанные достоинства метода прототипирования, многие разработчики игнорируют его. Как отмечает Нильсен, они мыслят следующим образом: *«Это слишком просто, чтобы могло работать»*. Однако для подобных идей нет никаких оснований. Напротив, различные исследования независимо друг от друга показывают, что тестирование с использованием прототипов высокой точности не дает сколько-нибудь серьезного выигрыша в количестве найденных проблем по сравнению с применением аналогов низкой точности (бумажных). При этом затраты на их создание заметно выше.

4.13. Контрольные вопросы

1. С какой целью проводится анализ рабочих заданий?
2. Как проводится сегментация пользовательской аудитории?
3. С какой целью создаются персонажи, и чем они отличаются от реальных пользователей?
4. Какие показатели измеряются в методе оценки производительности?
5. Назовите сильные и слабые стороны метода «мысли вслух».
6. Назовите особенности метода «конструктивное взаимодействие», отличающие его от метода «мысли вслух».
7. В чем заключаются функции модератора в методе «фокус-группа»?

8. Назовите особенности метода фокус-групп.
9. Какие вопросы проектирования интерфейса позволяет решить метод карточной сортировки?
10. Опишите процедуру проведения метода карточной сортировки.
11. В чем различие открытой и закрытой карточных сортировок?
12. Назовите участников метода эвристической оценки и процедуру его проведения.
13. Назовите 10 эвристик Я. Нильсена.
14. Дайте классификации прототипов, используемых в юзабилити-тестировании.
15. В каких ролях выступают участники тестирования в методе бумажных прототипов?
16. Какие проблемы выявляют с помощью метода бумажных прототипов?
17. Разработайте сценарий компьютерного варианта метода карточной сортировки.

4.14. Темы для дискуссии

1. Общее и различное в методах юзабилити-тестирования.
2. Перспективы автоматизации классических методов юзабилити-тестирования.

Глава 5.

Программно-аппаратные методы оценки юзабилити пользовательских интерфейсов

5.1. Принципы и методы регистрации движения глаз

Методы, использующие регистрацию движения глаз, давно привлекали интерес специалистов в области создания систем интерфейса в качестве перспективного метода управления динамическими объектами. На первый взгляд, кажется, что это — очень быстрый эффективный и простой метод фиксации намерений оператора. Однако довольно быстро в среде ученых и практиков пришли к пониманию сложной структуры зрительной деятельности. Результаты исследований движений глаз позволили сформировать общее представление о глазодвигательной системе как о следящей системе с переменными параметрами. Выделить направление взгляда оказалось довольно сложной процедурой.

Глаза почти всегда находятся в движении. Самому человеку кажется, что окружающий мир относительно неподвижен, а глаза движутся произвольно, но это не так. Даже когда мы пытаемся фиксировать взгляд на предмете или картине при их внимательном рассмотрении, глаза прекращают выполнять размашистые движения всего на 250–300 мс, а мелкие движения продолжают даже в этот период относительной неподвижности. Движения глаз выполняют функцию обеспечения новой информацией центрального (или фовеального) зрения, чтобы рецепторы не адаптировались и не прекращали передавать информацию в зрительную кору.

Движения глаз подразделяются на схватывающие, удерживающие и мелкие. Схватывающие движения направлены на поиск и фиксацию стимула. Основным движением этого типа является быстрое скачкообразное движение глаза — *саккада*. Ежедневно совершается приблизительно 230 000 саккадических движений глаз. Чем больше движение глаза, тем с большей скоростью оно совершается: при углах вращения глаза в 5° и меньше скорость движения составляет примерно $200^\circ/\text{с}$, а более крупные саккады в 100° и более могут достигать скорости, превышающей $700^\circ/\text{с}$. Вместе с тем, 85% саккадических движений глаз ограничены 15° . Считается, что это — баллистические движения: раз начавшись они должны достичь мишени прежде, чем инициируется следующее движение и направление изменится.

Саккады, по-видимому, чувствительны и адаптируемы к когнитивному входу и мишени или насыщенности зрительной сцены. Маловероятно, что последовательные саккады произойдут в промежутки времени меньше 150 мс, поскольку зрительной системе требуется около 40

мс на программирование саккады в процессе фиксации, от 20 до 30 мс — на выполнение саккадического движения и 50 мс — на восстановление четкого восприятия. Во время чтения саккады организованы в относительно упорядоченную последовательность, однако во время визуального поиска или разглядывания объектов они возникают без всякой системы.

Быстрая фаза оптокинетического нистагма также относится к схватывающим движениям. Когда человек следит за вращающимся полосатым барабаном или кружится на карусели, через определенные интервалы происходит резкий скачок глазного яблока с фиксацией взгляда на новом и обычно несколько смещенном назад (относительно вращения) элементе барабана или визуальной сцены.

Удерживающие движения направлены на сохранение схваченной сцены или предмета в области нейтрального (фовеального) зрения. В настоящее время выявлены три вида таких движений. Вестибулярные движения происходят в тех случаях, когда вестибулярная система внутреннего уха дает по цепи обратной связи команду глазодвигательной системе удерживать взгляд на мишени независимо от движений головы. Тихое преследование, слежение или медленная фаза нистагма — примеры удерживающего движения, при котором глаза следуют за движущимся стимулом. Если скорость движения стимула нарастает слишком быстро, инициируются корректирующие саккады, обеспечивающие повторный захват движущегося стимула для продолжения слежения за ним до тех пор, пока не будет достигнут следующий критический уровень корректировки. Вергентные движения тоже относятся к удерживающим, но выполняются они рассогласованно (глаза движутся в разных направлениях). К основным вергентным движениям относятся *конвергенция* и *дивергенция*, проявляющиеся в виде перекрытия зрительного поля. Вергентные движения можно продемонстрировать с помощью карандаша, удерживаемого прямо перед собой на расстоянии вытянутой руки от глаз: сгибая руку в локте и приближая карандаш к носу, мы вызовем конвергенцию, а удаляя карандаш от носа — дивергенцию глаз. Эти движения произвольны. Они совершаются медленно и обычно имеют малую угловую величину.

Третья группа чрезвычайно мелких движений известна под названием *физиологического нистагма*. Эти движения накладываются на более крупные движения и фиксации, обсуждавшиеся выше, и, по-видимому, имеют сходное происхождение, однако их назначение выяснить пока не удалось. Мелкий тремор имеет медианную амплитуду около 17 угловых секунд, а его частота составляет от 25 до 125 колебаний в секунду. Медленный дрейф происходит с амплитудой от 2 до 5 угловых минут и средней скоростью от 1 до 5 угловых минут в секунду. Это можно наблюдать,

поместив небольшой источник красного света в темной комнате. Фиксация взгляда на этом источнике света создаст впечатление его движения по комнате. *Микросаккады*, или рывки, являют собой другой вид физиологического нистагма. Они имеют амплитуду от 1 до 20 угловых минут и предельную скорость около $10^\circ/\text{сек}$.

Описывая методы регистрации движений глаз можно выделить две их основных группы: *контактные*, т.е. связанные с установкой регистрирующих датчиков непосредственно на роговицу глаза или вокруг него (электроокулография, фотооптический и электромагнитный методы), и *бесконтактные* (фотоэлектрический, кино и видео-регистрация).

В основе *метода электроокулографии* лежит использование собственных электрических свойств глазного яблока. По физической природе оно является диполем, в котором роговица относительно сетчатки электроположительна. Электрическая ось глазного яблока примерно совпадает с оптической осью и, следовательно, может служить индикатором направления взора. Изменение разности потенциалов между роговицей и сетчаткой обнаруживается через изменение потенциала в тканях, прилегающих к глазнице. Движения глаз регистрируются с помощью электродов, которые устанавливаются крестообразно вокруг глазной впадины.

Недостатки метода — невысокая разрешающая способность (точность дифференцировки мелких деталей $3^\circ\text{--}5^\circ$), достоинства — низкая стоимость оборудования. Регистрация не нарушает естественных условий зрительной активности и может выполняться как на свету, так и в темноте, даже с закрытыми глазами.

Фотооптический метод разработан А. Л. Ярбусом в 50-е годы XX века. Узкий пучок света, направленный на глазное яблоко, отражается от установленного на нем миниатюрного зеркальца и поступает на вход фоторегистрирующего устройства. Достоинства метода — высокая разрешающая способность, недостатки — необходимость жесткой фиксации головы испытуемого, контактный характер методики, регистрация возможна только в затемненном помещении. В настоящее время этот метод не используют.

Контактный электромагнитный метод обеспечивает столь же высокую разрешающую способность, но более удобен в плане регистрации. В основе метода лежит принцип изменения напряженности электромагнитного поля при изменении расстояния между излучателем и приемником. Излучатель крепят на глазном яблоке (с помощью центральной присоски, контактной линзы или кольца), приемные катушки располагают неподвижно вокруг головы испытуемого. Достоинства метода

— высокая разрешающая способность, основной недостаток — контактный характер методики.

Фотоэлектрический метод основан на преобразовании отраженного от роговицы пучка инфракрасного света в электрический сигнал. В настоящее время он не используется.

Кинорегистрация глаз известна с середины 60-х годов XX столетия, но из-за высокой трудоемкости не получила широкого распространения.

В последнее время, в связи с широким распространением персональных компьютеров и цифровых видеокамер, стала широко использоваться *видеорегистрация* движений. Глаз подсвечивается точечным источником инфракрасного излучения, а инфракрасная видеокамера производит скоростную съемку глаза.

На изображении программно определяется положение зрачка (в ИК лучах он представляет собой темный овал) и его размеры, а также позиция роговичного блика, представляющего собой отражение на роговице источника инфракрасного света. Направление взгляда система рассчитывает, основываясь на векторе, соединяющем позиции роговичного блика и центра зрачка. Достоинства методики — бесконтактный характер и возможность регистрации величины раскрытия.

5.2. Системы ай-трекинга

Возможность с высокой точностью измерять положение и анализировать движения глаз с определением направления взора — технология *отслеживания глаз, или ай-трекинга* — чрезвычайно важна для научных исследований, изучающих процессы зрительного восприятия или использующих зрительную стимуляцию, а также при проведении исследований для оценки эргономичности и последующего совершенствования интерфейса систем «человек-машина». Особое значение имеют исследования зрительной системы человека при анализе интерфейсов транспортных средств (самолета, автомобиля, поезда, корабля) в задачах управления воздушным движением, контроля электростанций, при изучении методов эффективного обучения операторов на тренажерах и обучающих системах (рис. 6).



Рис. 6. Система RED250 фирмы SensoMotoric Instruments GmbH (SMI) для дистанционного трекинга глаз

Технология отслеживания глаз также находит все больше применения в юзабилити-исследованиях операторских интерфейсов на этапе как проектирования, так и их эксплуатации и доработки (рис. 7).

Современные системы трекинга глаз обеспечивают

- высокую точность и надежность измерения движений глаз;
- монокулярные и бинокулярные измерения;
- эффективные алгоритмы анализа движений глаз;
- интеграцию со стандартными системами стимуляции (например, *Presentation, E-Prime, Superlab*);
- совместимость с другим исследовательским оборудованием;
- удобство настроек программного обеспечения, высокую скорость и эффективность управления аппаратурой;
- совместимость с другими психофизиологическими сенсорами (например, ЭЭГ, ЭКГ, КГР);
- решения для фМРТ, ЭЭГ, МЭГ и ПЭТ;
- интеграцию с тренажерами-имитаторами, реальными транспортными средствами и другим оборудованием;
- оценку статуса оператора и мониторинг внимания.

Технические характеристики систем:

- измерение горизонтального и вертикального положения взгляда, размера зрачка, моментов закрытия глаз и моргания;
- исследования микросаккад со скоростью регистрации до 1250 Гц;
- максимальное разрешение 0.01 градуса;

- непрерывный мониторинг, запись и передача в реальном времени данных о движении глаз;
- получение выходных данных в реальном времени в цифровом и аналоговом виде;
- интегрированная цифровая запись рассматриваемых сцен и точки зрения оператора на ней, функции просмотра;
- сетевой интерфейс для дистанционного получения потока данных и управления устройством;
- бесконтактные (или закрепленные на голове) системы отслеживания глаз;
- функции анализа данных (например, путь сканирования, анализ саккад, последовательность фиксационных точек, области интереса, карты внимания) как для отдельных испытуемых, так и для группы;
- экспорт данных в *ASCII* формат для последующего анализа статистическими программами (например, *MATLAB, SPSS, Excel*).

Области применения:

- психофизиология и психолингвистика;
- нейрофизиология;
- когнитивная нейробиология;
- маркетинговые исследования;
- анализ эргономичности;
- оценка действий оператора;
- обучение на виртуальных тренажерах.



Рис. 7. Методика фокусных карт для анализа видеофильмов с использованием ай-трекера

5.3. Основные понятия о работе трекеров движений глаз

Трекер оценивает точки взгляда с помощью датчика изображения, который находится перед глазами пользователя и рассчитывает точку взгляда по специальному математическому алгоритму. Иными словами, трекер имитирует наблюдение за взглядом человека для оценки его намерений и поведения (рис. 8).



Рис. 8. Ай-трекер с мобильным компьютером в качестве источника сканируемого рабочего поля

Большинство современных трекеров глаза основаны на фундаментальном принципе отслеживания отражений от роговицы одного или нескольких ближних ИК-излучателей, не видимых для человеческого глаза. На высокой скорости выборки один или несколько датчиков изображения регистрируют изображения глаз пользователя. Затем используют обработку изображения, чтобы найти точное положение бликов и определить правильное отражение и их точное местоположение. Математическую модель глаза используют для расчета «положения в пространстве и точки взгляда».

Перед исследованиями с помощью трекера проводят процедуру калибровки. Систему калибруют на глаза пользователя. Во время калибровки измеряют характеристики глаз пользователя, которые необходимы, чтобы правильно рассчитать направление взгляда. Используют 2, 5 или 9-точечную калибровку. Процедура занимает несколько секунд. Не требуются настройка оборудования или позиционирование пользователя. В некоторых трекерах можно использовать так называемую калибровку *free setup*, где она осуществляется незаметно для пользователя.

Современные ай-трекеры обеспечивают некоторую свободу движений головой в рамках воображаемого прямоугольника, в котором пользователь может двигаться и все еще быть отслеженным устройством. Трекеры

имеют различные диапазоны движения головы в зависимости от модели. Большой диапазон движений головы дает пользователю больше свободы перемещения и обеспечивает больший комфорт (рис. 9).



Рис. 9. Работа пользователя с трекером при изучении траекторий поиска информации на экране монитора

5.4. Компьютерные программы для проведения юзабилити-исследований

Юзабилити-исследования позволяют получить большие массивы разнородных и разнокачественных данных, которые необходимо обработать и сохранить для получения полезных рекомендаций при решении задач проектирования информационных систем. С этой целью в мире производят много систем для проведения экспериментов по эргономике, оценке интерфейсов, трекингу глаз и анализу результатов экспериментов. Они позволяют упростить основные этапы работы по использованию различных зрительных стимулов:

- подготовку сценария стимуляции, задающего последовательность предъявлений статических изображений, видеофильмов и *Web*-страниц с заданием времени экспозиции;
- автоматизированное проведение экспериментов, включающее калибровку, введение в базу данных об испытуемом, последовательное предъявление стимулов, сохранение результатов эксперимента;
- анализ данных с возможностью наложения результатов на использованный стимульный материал в режиме «проигрывания» эксперимента.

При сохранении результатов в базах данных производится их группировка на основе используемого сценария стимуляции. Это упрощает групповой анализ результатов.

В настоящее время используется ряд программных комплексов для решения задач юзабилити, из которых наиболее известна программа *Morae*, разработанная компанией TechSmith. Она позволяет организовать и реализовать все этапы юзабилити-исследования.

5.4.1. Общие сведения о программе *Morae*

Все участники тестирования взаимодействуют удаленно, по сети, работая каждый за своим компьютером. Модератор и наблюдатели удаленно наблюдают за действиями респондента. Модератор своевременно подает команды респонденту, наблюдатели в процессе тестирования делают пометки и комментарии.

Программа фиксирует последовательность действий респондента, записывает с помощью видеочамер копии экрана монитора и лицо респондента. Эта информация в режиме реального времени передается на компьютеры модератора и наблюдателей и параллельно записывается на жесткий диск в файл для последующей обработки.

Полученные данные — разнообразные потоки данных с компьютера респондента, комментарии наблюдателей — автоматически синхронизируются между собой. После завершения процесса тестирования данные легко обработать — в программе предлагается множество методов для обработки данных, автоматизации расчета количественных параметров оценки юзабилити, построения стандартных отчетов и графиков, создания презентаций и видеороликов на основе полученных данных.

Программный продукт *Morae* реализован в виде трех отдельных программ: *Morae Recorder* (для записи данных на компьютере респондента), *Morae Observer* (для удаленного наблюдения за действиями в режиме реального времени) и *Morae Manager* (для обработки полученных результатов и подготовки конфигурации процесса тестирования).

Morae Recorder

Программа *Recorder* устанавливается на компьютер респондента и ведет запись данных в процессе проведения тестирования. Для записи данных используется технология *Rich Recording Technology*, позволяющая записывать множество разнородных потоков данных в синхронизированном виде:

- экранное видео с компьютера респондента;
- видео с двух камер (одна снимает экран компьютера пользователя, другая — самого пользователя);
- аудиозапись голоса пользователя;

- информацию о событиях, происходящих за сценой (в приложениях и операционной системе);
- факты нажатия клавиш мыши и клавиатуры;
- комментарии, вносимые наблюдателями.

Morae Observer

Программа *Observer* позволяет наблюдателям удаленно подсоединяться к компьютеру респондента, наблюдать за его действиями, помечать моменты тестирования специальными маркерами и текстовыми комментариями в режиме реального времени. Наблюдатель может видеть рабочий стол респондента, просматривать видео с обеих камер, слышать голосовые комментарии. Несколько экземпляров программы *Observer* могут подсоединяться к одному и тому же экземпляру *Recorder*, т.е. за одним и тем же респондентом могут следить несколько наблюдателей одновременно.

Morae Manager

Процесс проведения тестирования настраивается модератором заранее. Для этого с помощью *Morae Manager* создается файл конфигурации, описывающий параметры процесса: список задач, которые должен выполнить пользователь, параметры записи данных на компьютере респондента, список маркеров, доступных наблюдателям, параметры использования автопилота, списки вопросов анкетирования. Один и тот же файл конфигурации может использоваться многократно при работе с разными респондентами — в этом случае *Morae* позволяет сравнить результаты, полученные от разных респондентов. В случае если конфигурация тестирования заранее не подготовлена, *Morae* предлагает ряд шаблонных конфигураций, ориентированных на тестирование сайтов, бумажных прототипов, мобильных устройств и т.д.

Morae Manager содержит интегрированный механизм анкетирования респондентов — после выполнения каждой задачи респонденту автоматически предлагается список вопросов, позволяющих ему оценить удобство работы с программой, простоту использования, степень удовлетворенности от работы с программой и т.п.

Тестирование может проводиться в ручном или автоматическом режимах. В ручном режиме работы процессом тестирования управляет человек — модератор. Он дает очередное задание респонденту, определяет момент, когда респондент должен отвечать на вопросы анкеты, и т.д. В автоматическом режиме работу модератора выполняет автопилот. Он последовательно предлагает пользователю задачи для выполнения и анкеты для заполнения. Респондент читает в окне автопилота очередную задачу, выполняет ее, нажимает кнопку «Задача завершена», получает

анкету, отвечает на нее, получает следующую задачу и т.д. — процесс полностью автоматизирован.

После того как тестирование завершено, все накопленные данные, включая данные с компьютера респондента и пометки наблюдателей, записываются в единый файл. Начинается следующая стадия процесса — обработка результатов.

Для изучения и обработки накопленных данных в программе *MoGae Manager* имеется вкладка «Analyze». Здесь можно просмотреть и прослушать записанные данные. На вкладке показывается текущий кадр видеоряда и единая временная шкала с нанесенными на нее маркерами и границами выбранной задачи. Шкала является масштабируемой — скорость воспроизведения видео можно изменять в широком диапазоне. Расположение маркеров и границы задач легко корректируются. Поиск нужного фрагмента данных на временной шкале не представляет проблем, благодаря средствам сквозного поиска. Например, можно просмотреть список моментов, в которые пользователь нажимал определенную клавишу на клавиатуре или, скажем, правую клавишу мышки при нажатой клавише «Ctrl», или вывести список маркеров определенного типа, внесенных конкретным наблюдателем.

Вкладка «Graph» позволяет создавать стандартные отчеты. *MoGae* поддерживает шесть типов отчетов — по работе в приложениях, по вводу данных, по маркерам, по заполнению анкет, по задачам и по веб-страницам. Для каждого отчета имеются несколько вариантов метрики. Например, для отчета по вводу данных в качестве метрики можно выбрать максимальное время между вводом данных, среднее количество щелчков мыши или движений мыши по задачам. Параметры метрики и диапазон обрабатываемых данных можно изменять. *MoGae* включает несколько стандартных шаблонных отчетов, которые можно создавать «одним щелчком» мышки. Есть возможность создавать собственные шаблоны отчетов. Созданные отчеты можно экспортировать в виде как картинок (EMF, BMP), так и таблицы данных (CSV).

Вкладка «Present» позволяет нарезать из исходного видеоматериала видеоклипы и создавать на их основе презентацию. Для этого вначале готовятся необходимые видеоклипы, графики и специальные картинки или клипы с комментариями, которые будут вставляться между видеоклипами и графиками. Затем создается раскадровка (storyboard) для презентации. Раскадровка отображается в виде набора кадров внизу окна. Далее в раскадровку помещаются все необходимые клипы и графики в нужном порядке. В завершение презентацию экспортируют в форматы файлов WMV, AVI или напрямую в PowerPoint.

5.4.2. Базовые характеристики систем обработки данных анализа движений глаз

Практика использования систем отслеживания движений глаз выделила следующие особенности организации систем трекинга и направления анализа данных:

- использование в качестве стимулов статических изображений, видеофильмов и *Web*-страниц;
- сохранение всех данных во встроенной базе;
- сравнение данных для нескольких экспериментов;
- экспорт графиков анализа и статистических данных;
- показ результатов анализа фиксации взгляда на экране, «температурные» карты;
- совместимость с монокулярными и бинокулярными данными;
- воспроизведение записанных перемещений взгляда;
- автоматическая детекция саккад, фиксационных точек и моргания.

В качестве примера системы реализующей указанные требования можно рассмотреть ряд программ фирмы *SMI*, предназначенных для систем мобильного трекинга:

- *eGaze 2 Video Package* используется для анализа видео, полученного с системы ай-трекинга. Обеспечивает расстановку и редактирование динамических зон интереса и последующий анализ по ним.
- *BeGaze 2 Reading Package* используется для анализа результатов экспериментов по чтению текста. Обеспечивает автоматическую генерацию множественных зон интереса по параграфам, предложениям, словам, буквам и последующий статистический анализ.
- *BeGaze 2 Observation Package* — модуль для записи и онлайн отображения видео и голоса испытуемого во время проведения эксперимента на основе дополнительной *Web*-камеры.

Все зрительные и аналитические результаты при анализе направления взгляда, перемещений глаз и параметров зрачка можно получить в различных удобных форматах. С помощью *BeGaze 2* легко провести

- анализ зрительного внимания на основе визуализации путей просмотра (*scanpath & bee swarm*), температурных карт, диаграмм фокуса внимания для статичных и динамичных стимулов;

- анализ зрительных паттернов поиска на основе задаваемых зон интереса; для систем *HED* (наголовные системы трекинга) используются динамические зоны;
- расчет ключевых индикаторов (*KPI*) с одновременным показом суммарной статистики для всех зон;
- количественный анализ саккад, фиксаций, морганий и времени реакции на внешние события;
- экспорт путей просмотра и карт внимания в фильмы или слайды;
- конфигурируемый статистический отчет с функциями группировки и сортировки;
- групповой анализ для нескольких испытуемых.

5.5. Типовой состав лаборатории юзабилити пользовательских интерфейсов

Для проведения юзабилити-тестирования не обязательно наличие специальной лаборатории, тестирование можно провести в официальной или неформальной обстановке: наиболее важно то, что вы выполните тестирование. Однако наличие лаборатории позволяет повысить качество и производительность юзабилити-тестирования, и в настоящее время многие организации, связанные с проектированием интерфейсов, имеют специализированные лаборатории. Основное назначение лаборатории юзабилити — проведение и обработка результатов широкого круга экспериментов по оценке пользовательских качеств интерфейсов и программных продуктов.

Под лабораторией юзабилити понимается комплекс программно-аппаратных средств, объединенный общими задачами исследования, направленного на регистрацию и оценку поведения пользователей при работе с проектируемыми или существующими интерфейсами.

Существуют два типа юзабилити-лабораторий: стационарные и мобильные (портативные). Они сильно различаются по своим возможностям, гибкости и стоимости оборудования — от ручной видеокамеры до специализированной юзабилити-лаборатории, содержащей последние разработки в области технологий записи и обработки мультимодальных данных.

Лаборатория представляет собой отдельное помещение, включающее рабочую зону (зону наблюдения) и диспетчерскую (зону контроля), разделенные звукопроницаемой перегородкой и полупрозрачным зеркалом Гезелла, которое позволяет исследователям скрытно наблюдать за испытуемыми (рис. 10, 11). Вход в рабочую зону и диспетчерскую

отдельный, что создает у испытуемых ощущение того, что они находятся в изолированном от наблюдения месте.

В рабочей зоне лаборатории расположены купольные видеокамеры и направленные микрофоны, позволяющие фиксировать поведение и реакции испытуемых на протяжении всего эксперимента и воспроизводить их с разных точек наблюдения при последующем анализе результатов. Используются дистанционно управляемые купольные камеры со встроенным панорамным управлением (перемещением) камеры и объективом с тихим *Zoom*-фокусом.

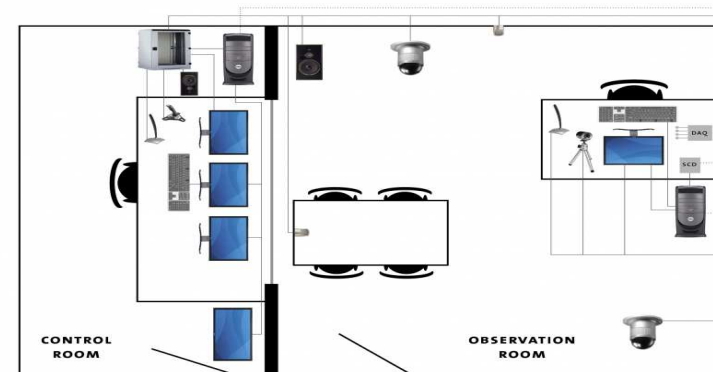


Рис. 10. Эскиз плана юзабилити-лаборатории



Рис. 11. Внешний вид типовой юзабилити-лаборатории

Купольные камеры используют матрицы с высокой чувствительностью, что дает возможность производить эксперименты при пониженном освещении, что важно при оценке мобильных устройств, работающих в разных условиях освещения.

Запись звука осуществляется с помощью чувствительных микрофонов, которые установлены высоко над стеклом со стороны рабочей комнаты и перед испытуемыми. Эти микрофоны обеспечивают высокое качество звука, не требуя от участников ношения нагрудных микрофонов.

Общение между испытуемыми и экспериментаторами осуществляется посредством системы громкоговорящей связи.

Типичный комплект аудио-видеооборудования *стационарной юзабилити-лаборатории* включает в себя

- одну или несколько видеокамер для записи поведения участников исследования (например, выражения лица, руки, движения, позы);
- преобразователь развертки для съемки (видео) экрана компьютера, с которым взаимодействует участник тестирования;
- видео-микшер для комбинирования этих источников видеосигнала в разбитом на зоны экране (*split-screen*) и с врезкой (*picture-in-picture*) изображения «картинка в картинке»;
- один или несколько микрофонов, чтобы записать голоса участников и экспериментаторов;
- аудио-микшер, для объединения этих аудио-входов и связи, чтобы обеспечить обмен между участниками тестирования, руководителем тестирования и другими наблюдателями;
- видео-рекордер для хранения и вывода видео, аудио-микшер;
- один или более видеомониторов для отображения этих источников видео;
- один или более громкоговорителей с усилителями для наблюдателей и испытуемых с воспроизведением звука в реальном времени;

Компьютерное оборудование может включать компьютер, с которым взаимодействует выполняющий тест участник, и отдельный компьютер для использования экспериментаторами или для регистрации данных другого наблюдателя.

Все возможности стационарных лабораторий теперь доступны в *портативных устройствах*, которые могут легко поставяться и быстро разворачиваться в рабочее состояние на удаленных объектах.

Существуют различные комплекты оборудования, подходящие для портативных лабораторий. Все они предназначены для высокой интеграции аудио-видеокomпонентов, что сводит к минимуму время настройки аппаратуры и обеспечивает максимальную долговечность.

Портативные лаборатории, как правило, включают один или несколько маленьких видеомониторов, но могут иметь выходы для подключения больших мониторов, если они доступны в тестовой среде. За исключением относительно небольших мониторов портативные лаборатории имеют вполне современное и функциональное оборудование, требуемое и достаточное для проведения юзабилити-исследований.

Стационарные лаборатории часто включают в себя следующее оборудование, обеспечивающее дополнительные возможности:

- видеокамеры, установленные на стенах или потолке: иногда камеры, поддерживающие функции панорамирования, наклона и зума (*pan-tilt-zoom*), чтобы следовать за участниками движения;
- смежные рабочие места для участников тестирования и наблюдателей со смотровым порталом, оснащенным дополнительным зеркалом из полупрозрачного стекла;
- большой видеомонитор или проекторы для размещения групп наблюдателей;
- вспомогательный монитор компьютера для наблюдателей, зеркало, компьютер, монитор просмотра для участника;
- дополнительные видеомониторы, позволяющие вести одновременную многоканальную запись/воспроизведение в реальном времени;
- оборудование для редактирования видеооборудования для производства спецэффектов;
- вспомогательная клавиатура и мышь, чтобы контролировать участника с компьютера из соседнего помещения.

Преимущества стационарной лаборатории включают в себя следующее:

- Она имеет отдельную комнату для наблюдателей. Это позволяет иметь больше участников юзабилити-тестирования, чем если бы все находились в одной комнате. Иметь комнату для наблюдателей важно потому, что в юзабилити-тестировании большое влияние на результаты теста оказывает непосредственный контакт с экспериментатором;

- В стационарной юзабилити-лаборатории проще провести официальную процедуру юзабилити-тестирования. Официальное юзабилити-тестирование связано с получением количественных документированных данных, которые могут использоваться в задачах экспертизы изделий и иметь в случае необходимости юридическую силу;
- Даже если вы делаете неофициальное качественное тестирование удобства использования продукта, вы можете сделать запись сессии;
- Возможность распределения обязанностей между сотрудниками и специализация их, что позволяет повысить скорость выполнения и качество работы;
- Стационарная юзабилити-лаборатория оказывает мощное воздействие на заказчиков, напоминая им о важности тестирования удобства использования продукта. Иногда даже внешний вид юзабилити-лаборатории с ее помещениями, оборудованием и т. д. дает понять клиенту, что юзабилити требует серьезных усилий.

Преимущества портативной лаборатории включают

- независимость от помещений и расписания работы;
- отсутствие оплаты аренды помещения;
- естественность условий деятельности испытуемых;
- неформальность обстановки проведения юзабилити-исследования, ненужность дополнительных формальных согласований;
- гибкость варьирования условий и содержания экспериментов.

5.6. Контрольные вопросы

1. Дайте классификацию движений глаз и роль каждого вида движений в процессах приема зрительной информации.
2. Дайте характеристику удерживающих движений глаз в процессе информационного поиска.
3. Что такое «физиологический нистагм»?
4. Какие методы используются для регистрации движений глаз?
5. Назовите сильные и слабые стороны метода электроокулографии.
6. В чем состоит сущность фотооптического метода Ярбуса?
7. В чем заключаются особенности метода кинорегистрации глаз?
8. Опишите принцип действия трекера движений глаз.

9. Какие вопросы в процедуре юзабилити-тестирования решаются с помощью специализированных компьютерных программ?
10. Опишите функции программного пакета *Morae* и его состав.
11. Какие задачи решаются с помощью программы *Morae Manager*?
12. Какими особенностями в организации данных обладают системы отслеживания движений глаз?
13. Цель и задачи юзабилити-лаборатории.
14. Приведите минимальный набор средств для проведения юзабилити-исследования.
15. В чем сходство и различия между стационарными и мобильными юзабилити-лабораториями?
16. Что такое «зеркало Гезелла» и его роль в юзабилити-исследованиях?
17. Нарисуйте план типовой юзабилити-лаборатории и дайте характеристики ее элементов.
18. Опишите типовой состав оборудования юзабилити-лаборатории.
19. Какими преимуществами обладает стационарная юзабилити-лаборатория по сравнению с мобильной?

5.7. Темы для дискуссии

1. Роль методов Ай-трекинга в юзабилити. Области применения.
2. Разработайте план и состав программно-аппаратных средств для юзабилити-лаборатории по исследованию и проектированию веб-сайтов. Обоснуйте применение элементов оборудования лаборатории.
3. Тенденции и направления развития технологий юзабилити-тестирования.

Глава 6.

Рекомендации по повышению юзабилити компьютерных пользовательских интерфейсов

Компьютерные интерфейсы должны отвечать требованиям юзабилити, инженерной психологии и эргономики. Юзабилити — степень, в которой продукт может быть использован определенными пользователями, в определенном контексте использования, для достижения определенных целей, с должной эффективностью, продуктивностью и удовлетворенностью. Применительно к компьютерной технике термином *юзабилити* называют концепцию разработки пользовательских интерфейсов программного обеспечения, ориентированную на максимальное психологическое и эстетическое удобство для пользователя.

Юзабилити — это совокупность свойств инструмента, влияющих на эффективность его использования в конкретной предметной деятельности и выражающихся в применимости данного инструмента, легкости его освоения и использования, воспроизводимости полученных навыков, в низкой частоте ошибок, в субъективном удовольствии и прочих факторах.

В настоящих рекомендациях будут приведены начальные сведения о юзабилити программных интерфейсов, включая философию юзабилити-проектирования и некоторые рекомендации по оформлению графических слайдов.

6.1. Философия дизайна компьютерного интерфейса

Под философией понимаются общие взгляды и принципы, на основании которых производится проектирование компьютерного интерфейса, его содержание и оформление. Оформление интерфейса должно отвечать основным требованиям юзабилити. Разработчик интерфейса должен использовать метафоры, психические модели, различать явные и подразумеваемые действия пользователя, обеспечивать согласованность, обратную связь, манипуляции с учебным содержанием и т. д.

6.1.1. Использование метафор

Вспользуйтесь знаниями людей о мире, используя метафоры для передачи понятий и особенностей разрабатываемого вами интерфейса. Метафоры являются строительными блоками в ментальной (умственной) модели пользователя. Использование метафор, которые представляют собой конкретные, знакомые идеи, позволяет пользователям использовать для обучения знакомые им модели и понятия, что повышает

эффективность учебного процесса. Например, используя метафору папок для хранения документов люди могут организовать свои жесткие диски аналогично тому, как они организуют картотеки. Метафора корзины помогает удалять ненужную информацию с рабочего стола и т. д.

Метафоры нужно предлагать для определенного элемента, но их использование не должно ограничивать их осуществление. Важно найти баланс между предложенной метафорой и способностью компьютера поддерживать и расширять её. Например, количество элементов, которые пользователь помещает в корзину не должно ограничиваться количеством элементов физической мусорной корзины. Это уже есть ментальная модель, описывающая решение задачи с помощью интерфейса. Эта модель возникает из комбинации реальных переживаний человека, эрудиции, его опыта работы с другим программным обеспечением (ПО), а также с компьютерами в целом. Например, пользователи имеют реальный опыт написания и рассылки электронных писем и используют приложения электронной почты, чтобы написать и отправить письмо. Исходя из этого, пользователь имеет концептуальную модель этой задачи, которая включает в себя определенные ожидания, такие как способность создавать новое письмо, выбрать получателя и отправить письмо. Приложение электронной почты, которое игнорирует ментальную модель пользователей и не отвечает, по крайней мере, некоторым их ожиданиям, будет для них трудным и даже неприятным в использовании. Это же проявляется при использовании незнакомых для пользователей концептуальных моделей, без опоры на имеющиеся у них знания и опыт.

Прежде чем создавать пользовательский интерфейс вашего приложения, попробуйте открыть для себя ментальную модель задачи ваших пользователей, которую ваше приложение помогает им выполнить. Будьте осведомлены о присущих модели метафорах, которые представляют собой концептуальные компоненты задачи. Например, в метафоре письма содержатся буквы, почтовые ящики и конверты. В ментальной модели задачи, связанной с фотографией, метафоры включают в себя фотографии, фотоаппараты и альбомы. Стремитесь отражать ожидания пользователя по отношению к компонентам задач, организации рабочего процесса имеющимися в вашем распоряжении оконными меню и панелями инструментов.

Ментальные модели, имеющиеся у ваших пользователей, должны наполняться дизайном пользовательского интерфейса. Он должен сообщить раскладку окон приложения, подбор и организацию икон и элементов управления в панелях инструментов и функциональность панелей. Кроме того, вы должны поддерживать ментальную модель пользователя, стремясь включить следующие характеристики:

— *Знакомость.* Ментальная модель пользователя базируется прежде всего на опыте. По возможности нужно усилить компоненты пользовательского интерфейса, чтобы отразить символику модели и отображение меток, которые используют терминологию модели. Затем необходимо использовать привычные *Windows*-компоненты пользовательского интерфейса, которые предлагают стандартные функциональные возможности, такие как поиск и навигацию иерархических наборов данных.

— *Простота.* Ментальные модели задачи, как правило, упорядочены и сосредоточены на основных компонентах задачи. Хотя может быть множество дополнительных деталей, связанных с той или иной задачей, однако основные компоненты не должны конкурировать с деталями за внимание пользователя.

— *Доступность.* Следствием простоты является доступность. Незагроможденный пользовательский интерфейс необходим, но в равной мере необходимо и наличие определенных ключевых функций и настроек. Избегайте скрывать такие компоненты слишком глубоко в подменю или делать их доступными только из контекстного меню.

— *Возможность делать открытия.* Поощряйте пользователей возможностью открыть для себя функциональность вашей программы, обеспечивая сигналы о том, как использовать те или иные элементы пользовательского интерфейса. Если элемент является интерактивным (например окно), попробуйте нажать его. Хорошо разработанные иконки панели инструментов делают команды узнаваемыми для пользователей. Это дает пользователям уверенность в изучении функциональных возможностей нового для них приложения.

6.1.2. Явные и подразумеваемые действия

Каждая операция в программе включает в себя манипуляции объекта с помощью действий. На первом этапе этой манипуляции пользователь видит нужный объект на экране, на втором — он выбирает или назначает этот объект, на заключительном этапе он выполняет действия либо с помощью команд меню, либо путем прямого манипулирования над объектом с помощью мыши или другого устройства. Это приводит к двум парадигмам для манипулирования объектами: *явные* и *подразумеваемые* действия.

Явные действия связаны с четким результатом манипулирования объектом (например, меню со списком команд, которые могут быть выполнены над выбранным объектом). Название команды меню ясно указывает на то, что действие и текущее состояние команды (серым цветом или включено) показывают, имеет ли действие место в данном

контексте. Явные действия не требуют от пользователя запоминания команд, которые могут быть выполнены над данным объектом.

Подразумеваемые действия передают результат действия через визуальные сигналы или контекст. «Перетаскать» — операция перетаскивания является распространенным примером подразумеваемого действия. Перемещение одного объекта на другой представляет собой отношение между объектами и действиями, которые должны выполнять операция перетаскивания. Например, перетаскивание значка файла в корзину предполагает неизбежное удаление основного файла из файловой системы. Для подразумеваемых действий, которые будут очевидны, пользователь должен быть способен распознавать задействованные объекты, манипуляции, которые должны быть выполнены, и представлять себе последствия действий. Держите эти две парадигмы в виду при проектировании пользовательского интерфейса учебной программы, изучите мысленную модель пользователя. Это поможет вам определить, где, когда и какой тип действия является целесообразным.

6.1.3. Прямая манипуляция

Прямая манипуляция подразумевает действие, которое позволяет пользователям почувствовать, что они контролируют объекты, представленные на компьютере. Согласно этому принципу экранный объект должен оставаться видимым, в то время как пользователь выполняет действия с ним, и влияния действия должны быть немедленно видны. Например, с помощью операции перетаскивания (наиболее распространенный пример прямых манипуляций) пользователи могут переместить файл, перетаскив его значок из одного места в другое, или перетаскать выделенный текст прямо в другой документ. Другие примеры прямого манипулирования — изменения размера графического объекта в приложении типа «рисунок» и расположение объекта или камеры в трехмерной сцене.

Поддерживайте прямые манипуляции, когда есть вероятность того, что пользователи их ожидают. Избегайте заставлять пользователей использовать элементы управления прямой манипуляции для работы с данными.

На рабочем столе пользователем выполняется выбор из альтернатив, представленных на экране. Пользователи взаимодействуют непосредственно с экраном, выбирая объекты с помощью указательного устройства, как правило, мыши, чтобы она указывала на элементы, расположенные на рабочем столе.

Windows рабочий стол работает по двум основным парадигмам, которые базируются на двух основных предположениях:

- пользователи могут видеть на экране то, что они делают;
- пользователи могут указывать на то, что они видят.

Парадигмы основаны на общем виде действия пользователя: существительное — глагол — действие. В первой парадигме пользователь выбирает интересующий объект (существительное), а затем выбирает действия, выполняемые на объекте (глагол). Все действия, доступные для выбранного объекта, перечислены в меню, так что пользователи, которые не уверены, что им делать дальше, могут освежить это в памяти путем сканирования по меню. В любой момент пользователь может выбрать любое доступное действие без необходимости запоминать какую-либо конкретную команду. Например, пользователь щелкает значок документа (существительное), а затем печатает (глагол) документ, выбирая «Печать» из меню «Файл».

Во второй парадигме пользователь перетаскивает объект (существительное) на другой объект, который имеет действие (глагол), связанное с ним. На рабочем столе, например, пользователь может перетаскивать иконки в корзину. Пользователь не выбирает действие из меню, но ясно, что произойдет в данном случае с одним объектом, когда он расположен на другом объекте. Например, перетаскивание значка документа в корзину означает удаление этого документа, а перетаскивание на значок принтера — печать документа.

Нельзя давать неопределенную форму действию и требовать от пользователя решение логической задачи. Действия должны быть однозначно интерпретируемыми.

6.1.4. Обратная связь и коммуникации

Обратная связь и коммуникации включают в себя информирование пользователей о том, что происходит, предоставляя им соответствующую информацию.

Когда пользователь инициирует действие, всегда необходимо обеспечивать для него признак того, что ваше приложение получило пользовательский ввод и работает на нем. Пользователи хотят знать, что их команда выполняется. Если команда не может быть осуществлена, они хотят знать, почему и что можно сделать вместо этого.

Анимация, если использовать ее с осторожностью, является одним из лучших способов, чтобы показать пользователю, что требуемое действие осуществляется. Вы можете использовать анимацию для того, чтобы прояснить отношения между объектами и последствия действий.

Например:

- когда пользователь минимизирует размеры окна, оно не просто исчезнет, а плавно соскользнет в меню, ясно говоря пользователю, где можно найти его снова;
- когда показываются отношения между листом и окном, лист разворачивается из заголовка окна;
- чтобы подчеркнуть связь между иконкой ящика и окном, ящик выдвигается из-под окна, отображая затенение, что делает его похожим на ящик стола.

Следует по возможности использовать тонкие эффекты анимации, так как это повышает уровень обратной связи в пользовательском интерфейсе.

Для потенциально длительной операции используйте индикатор, предоставляющий полезную информацию о том, как долго будет продолжаться эта операция. Пользователям не нужно точно знать, сколько секунд операция займет, но оценка всегда полезна.

Необходимо обеспечить прямую простую обратную связь, чтобы люди могли ее понять. Например, в сообщениях об ошибках должна быть точно изложена ситуация, вызванная ошибкой («Не хватает места на диске для сохранения документа»), и возможные действия пользователя, которые можно предпринять, чтобы исправить ее («Попробуйте сохранить документ в другом месте»). Если ваше приложение состоит из процесса переднего плана, который отображает пользовательский интерфейс, и фоновый процесс, который выполняют все или некоторые из основных задач приложения, нужно проявлять особую осторожность, чтобы провести все коммуникации с пользователем с помощью пользовательского интерфейса на процессе переднего плана. В частности, фоновый процесс никогда не должен отображать диалоговое окно или окно, в котором пользователю необходимо изменить настройки или предоставить информацию. Если фоновый процесс должен общаться с пользователем, он должен быть выдвинут на передний план применения. Это важно, поскольку пользователь может не знать (или забыть), что фоновый процесс запущен, и получение сообщения об этом могло бы привести к путанице.

6.1.5. Согласованность в интерфейсе

Последовательность в интерфейсе позволяет пользователям передавать свои знания и навыки из одного приложения в другое. Используйте стандартные элементы интерфейса *Windows* для обеспечения согласованности внутри вашего приложения и получения выгоды от согласованности между приложениями.

Задавайте себе вопросы:

— Соответствует ли работа вашего приложения ожиданиям пользователей?

— Удовлетворены ли потребности пользователей без дополнительных операций?

— Соответствует ли приложение ментальной модели пользователя? (соответствие ожиданиям каждого конкретного пользователя является самым сложным видом согласованности). Особенно этого трудно достигнуть, если приложение будет использоваться аудиторией с широким диапазоном знаний. Вы можете решить эту проблему, тщательно взвешивая последовательность решаемых вопросов в контексте вашей целевой аудитории и их потребности.

6.1.6. «Что вижу, то и получаю»

В приложениях, в которых пользователи могут форматировать данные для печати, публиковать в Интернете или записывать на пленку, DVD или в других форматах, нужно убедиться в том, что нет никаких существенных различий между тем, что пользователи видят на экране и тем, что они получают в конечной продукции. Используйте, если это необходимо, функции предварительного просмотра. Пользователи должны иметь возможность найти все доступные функции, имеющиеся в вашем учебном курсе. Не скрывайте функции, если они доступны в меню. В меню должны быть представлены настоящие списки команд, так что пользователи могут их видеть и выбирать, а не пытаться вспомнить имена команд. Избегайте предоставления доступа к функциям только в панели инструментов или контекстном меню. Команды всегда должны быть доступны для пользователя и не требовать значительных усилий памяти для поиска места их хранения.

6.1.7. Обеспечение возможности ошибаться

Поощряйте пользователей к изучению вашего интерфейса путем прощения их ошибок, т.е. делая большинство действий легко обратимыми. Пользователи должны чувствовать, что, если они попытаются сделать что-либо, это не приведет к повреждению системы и потере их данных. Эффективно создание механизмов защиты, таких как применение команд «Отменить» и «Вернуть к сохраненному», так что пользователи будут чувствовать себя комфортно при изучении и использовании вашего интерфейса.

Предупреждайте пользователей, когда они иницируют задачу, которая приведет к необратимой потере данных. Однако если предупреждения появляются слишком часто, то это может означать, что

продукт имеет недостатки. Когда варианты изложены ясно и обратная связь является своевременной, используемое приложение должно быть относительно свободным от ошибок.

Необходимо предвидеть общие проблемы и предупреждать пользователей о возможных побочных эффектах. Необходимо также предоставлять обширную обратную связь и коммуникации на каждом этапе, так чтобы пользователи считали, что у них есть достаточно информации, чтобы сделать правильный выбор.

6.1.8. Воспринимаемая устойчивость интерфейса

Интерфейс предназначен для обеспечения понятных, знакомых и предсказуемых условий. Чтобы дать пользователям визуальное ощущение стабильности, интерфейс определяет множество стандартных графических элементов, таких как меню, окна управления и т.д. Эти стандартные элементы предоставляют пользователям знакомую среду, в которой они знают, как вещи ведут себя и что можно и нужно делать с ними.

Чтобы дать пользователям в концептуальном смысле стабильность, интерфейс должен обеспечивать ясное, конечное множество объектов и набор действий над ними. Например, когда команда меню не распространяется на выбранный объект или объект в его текущем состоянии, команда должна быть недоступна. Чтобы передать ощущение стабильности, можно вводить сохранения пользователем изменяемых параметров, таких как размеры окна и местоположения. Когда пользователь устанавливает их, то экранная среда должна иметь определенное расположение; параметры должны оставаться такими же, пока пользователь не изменяет их.

Предоставление информации о текущем состоянии приложения и обратная связь способствуют восприятию его стабильности, позволяя пользователям знать, что приложение выполняет нужную задачу.

6.1.9. Эстетическая целостность

Эстетическая целостность означает, что информация хорошо организована в соответствии с принципами хорошего визуального дизайна. Ваш продукт должен выглядеть приятным на экране, даже при его просмотре в течение длительного времени

Используйте графику только тогда, когда она действительно повышает удобство использования. Не перегружайте окна и диалоги десятками значков или кнопок. Не следует использовать условные обозначения для представления концепций: они могут запутать или отвлечь пользователей. Общая компоновка окна и дизайна элементов

пользовательского интерфейса должна отражать ментальную модель выполняемой пользователем задачи.

Есть ряд полезных приемов, которые вы можете применить для обеспечения высокого эстетического качества интерфейса:

- все иконки должны быть представлены с максимальным графическим качеством;
- все тексты, использующие стандартные шрифты в системе, должны быть сглаженными автоматически;
- размер шрифта и тип должны соответствовать друг другу в пределах окна;
- контроль размеров должен соответствовать в границах окна;
- не следует смешивать малые и стандартные элементы управления;
- нельзя изменять значение или поведение стандартных элементов (например, всегда используйте флажки для нескольких вариантов, но не для взаимоисключающих вариантов);
- используйте кнопки для немедленного исполнения таких команд, как «открыть»;
- не используйте кнопки для отображения всплывающих меню;
- не используйте иконические кнопки в виде вкладок.

6.1.10. Совместимость интерфейсной среды с операционной системой

Чтобы убедиться, что ваше приложение «чувствует себя как дома» в среде операционной системы *Windows*, соблюдайте следующие правила:

1. Избегайте отягощения пользователей приложениями, в которых решаются задачи управления. Стоит подчеркнуть очевидный факт, что пользователи просматривают приложения иначе, чем вы. Нигде эта разница не будет более яркой, чем при выполнении общих приложений задач управления, таких как поиск и открытие документов, открытие и закрытие окон. Хотя есть много способов позволяющих сделать такие задачи проще для пользователей, более важный вопрос: почему пользователи не выполняют их так же, как все?

2. Воспользуйтесь возможностью сделать шаг назад и пересмотреть разделение труда между пользователями и приложениями. Например, если ваша программа представляет собой однооконное приложение, которое не выполняет фоновые задачи, задайте себе вопрос: нужно ли пользователю явно выйти из вашего приложения после закрытия окна? На самом деле, если пользователю все равно, как ваше приложение в данный момент

выполняется, его окна могут быть закрытыми, либо расположенными на другом месте рабочего стола. Изменения в том, как пользователи просматривают состояния и расположение их содержания, должны заставить вас задуматься о том, как ваше приложение представляет задачи, которые связаны с этими понятиями.

3. Насколько возможно восстановите предыдущее состояние пользователя. Пользователи не должны помнить, какие окна были открыты (и которые были открыты на весь экран), когда они входят в систему или начинают свою работу с компьютером. Для опытных пользователей нужно дать возможность понять, где они были в момент прерванного вашего приложения. Если пользователи могут создавать документы в вашем приложении, не забудьте ввести функцию автоматического сохранения *Auto Save*.

4. Избегайте обращения внимания пользователя на форматы файлов. Лучше всего, когда пользователи не должны думать о форматах файлов. Кроме того, пользователи, как правило, ожидают, что смогут открыть другие документы в вашем приложении и поделиться с другими документами, которые они создают в своем приложении. Не забудьте указать имя и расширение файла, соответствующие содержанию документа. В то же время соблюдайте меры по сохранению пользовательских предпочтений при отображении расширения, имен файлов и документов в своем приложении.

6.1.11. Соответствие функций интерфейса достигаемым целям

Когда люди используют ваш интерфейс, они делают это не для изучения его возможностей, а для достижения своих целей. Чтобы убедиться, что ваше приложение позволяет людям достичь целей, самый простой способ — следить за тем, чтобы каждая функция тесно интегрировалась с решением, которое вы предоставляете пользователю.

Избегайте введения каскада функций. Может быть очень заманчивым добавление в интерфейс функций, которые не полностью имеют отношение к основным направлениям вашего приложения, но это может привести к раздутому интерфейсу, который будет медленным, сложным и трудным в использовании. Всегда спрашивайте себя, поддерживает ли предлагаемая функция непосредственно цели пользователя, и если нет, то избегайте ее.

Соблюдайте правило 80–20, которое гласит, что примерно 80 % пользователей используют только часть функций приложения, в то время лишь около 20 % используют все (или почти все) функции. Анализ аудитории позволяет вам подчеркнуть особенности программного интерфейса, которые решают главные задачи.

6.1.12. Простота в использовании

Простое в использовании приложение предлагает хорошо продуманный, интуитивно понятный интерфейс, который поддерживает элегантные решения сложных проблем. Кроме того, простые в использовании приложения предоставляют пользователям инструменты, которые актуальны в конкретных условиях, ликвидируя или отключая не имеющие значения опции.

Когда это целесообразно, необходимо использовать метафоры, которые представляют собой конкретные, знакомые идеи. Соответствующие метафоры позволяют людям использовать имеющиеся знания и опыт для использования вашего приложения. Обязательно используйте метафоры, которые соответствуют задачам в вашем приложении. Если у вас в программе есть возможность исказить метафору, то вы, скорее всего, запутаете людей вместо того, чтобы им помочь.

Убедитесь, что в описании вашего приложения ясно сказано, что нужно пользователям, чтобы использовать его. Это относится также и к любой другой вами поставляемой версии продукта. Пользователи должны знать системные требования к вашему приложению до его установки, потому что они могут начать использовать его сразу же по получении.

В первую очередь представляется наиболее популярным использование *прогрессивного раскрытия*. Техника прогрессивного раскрытия состоит из сокрытия дополнительной информации или более сложных функций пользовательского интерфейса, пока нет пользовательских потребностей или по его просьбе. Использование прогрессивного раскрытия может улучшить ваше приложение двумя способами: оно помогает начинающим пользователям понять ваше приложение, не будучи перегруженными информацией или функциями, которые им не нужны, и в то же время дает более опытным пользователям доступ к продвинутым функциям, которые они хотят использовать.

Когда это целесообразно, необходимо использовать панели для упрощения пользовательского интерфейса. Незагроможденный пользовательский интерфейс является желательным, но наличие определенных ключевых функций и настроек для пользователя в нем должно быть. Панель может быть хорошим способом для представления часто необходимых функций контроля параметров, влияющих на текущий документ или выбор.

Поскольку панели остаются открытыми, пока пользователи не закрывают их, важно взвесить их полезность по отношению к занимаемому ими экранному пространству. Насколько возможно, нужно организовать меню так, чтобы оно отражало иерархию объектов в приложении. Например, если ваше приложение помогает пользователям

создавать компьютерную анимацию, приложениями конкретного меню могут быть декорации, персонажи, фоны и проекты. На более высоком личностном уровне проекта он содержит сцены, каждая из которых содержит фоны и персонажи; естественным порядком этих меню могут быть проекты, сцены, фоны и персонажи.

В общем случае в меню интерфейса отображаются команды для обработки высшего уровня, располагаются более универсальные объекты на одном конце строки меню и более конкретные лица — на противоположном конце. Не рекомендуется использование динамических элементов меню, лучше просто сократить меню приложения. Динамическое меню (т.е. пункт меню, который меняется, когда пользователь нажимает клавишу-модификатор) может быть хорошим способом дать готовые ответы для искушенных пользователей, но его поиск может составлять трудности для неопытных пользователей. В целом лучше перечислить все важные команды в меню, даже если это делается немного дольше.

Важно показать информативное, действенное предупреждающее сообщение, когда что-то пойдет не так. Предупреждение должно четко передать то, что случилось, почему это произошло, и варианты к юридическому разбирательству. Опишите решения проблемы, если таковая имеется, и сделайте все, что вы можете, чтобы не дать пользователю потерять данные. Старайтесь не использовать оповещения для предоставления информации о том, что пользователь не может действовать.

Используйте отображаемые имена на месте сырых путей и имен файлов. Это поможет пользователям не быть подверженными влиянию используемого в приложении способа работы с файлами и данными. Не забудьте обратить внимание на установленный язык пользователя и предпочитаемые расширения файлов, так что вы можете отображать их содержание в соответствии с пользовательскими ожиданиями.

6.1.13. Качественная графика

Качественная графика не только улучшает внешний вид вашего интерфейса, она также помогает донести информацию до пользователей и улучшить впечатление от программного продукта в целом.

Убедитесь, что ваша графика выглядит профессионально. Не стоит недооценивать влияние красивой, высококачественной графики на ваших пользователей. Как никогда ранее, люди любят видеть великолепную, профессиональную графику вокруг них, и большинство людей легко распознает различия в качестве. Низкое качество графики создает плохое

впечатление и может негативно повлиять на восприятие общего качества приложения.

Чтобы убедиться в том, что пользователи в восторге от вашего приложения, дайте графике приоритет в вашем процессе проектирования и разработки. Убедитесь, что ваша графика отлично смотрится в полноэкранном режиме. Как правило, всегда начинайте с большего, чем нужно, объема художественных работ и затем уменьшите его. Когда это уместно, рассмотрите вопрос о включении изображения реальных материалов. В некоторых случаях использование существующих в реальном мире текстур (таких как дерево, кожа, металл, бумага) может повысить качество приложения и придать дополнительный смысл для пользователей.

Если вы хотите узнать, имеет ли смысл в вашем приложении создание тектуры, убедитесь в том, что она улучшает работу пользователей и понимание. Создайте хорошее первое впечатление о программе с красивой иконки приложения. Значок вашего приложения дает пользователям первый опыт работы и может оказать заметное влияние на их ожидания. Подумайте о значке своего приложения как о вашей визитной карточке и не жалейте ресурсы, необходимые для того, чтобы произвести хорошее впечатление на пользователей. Насколько возможно, следует использовать стандартные иконки в своем приложении, так как пользователи уже знают, что они означают. Если вам нужно представлять действие, для которого не существует стандартного значка, необходимо создать собственный значок, который имеет четко выраженное значение для пользователей.

6.1.14. Скорость отклика системы на действия пользователя

Оперативность является критерием, с помощью которого пользователи измеряют производительность программного обеспечения. Ваше приложение может использовать лучшие алгоритмы обработки данных и доступные методы повышения производительности, но если оно не мгновенно реагирует на пользователя, оно будет казаться медленным.

Во время разработки приложения нужно уделять внимание факторам, влияющим на восприятие пользователя, и фактически измеримые показатели производительности вашего приложения. Необходимо мгновенно опознавать команду пользователя и ее ввод. Пользователи ожидают получить какой-то конкретный тип обратной связи каждый раз, когда они взаимодействуют с вашим приложением, (например, выделение кнопок, когда пользователи щелкают на них, и указатели, изменяющиеся соответствующим образом, когда пользователи перемещают различные элементы управления).

Аналогичным образом, если команда не может быть осуществлена, пользователи хотят знать, почему она не может быть исполнена? Что они могут сделать вместо этого? Чем быстрее вы предоставляете обратную связь для взаимодействия с пользователем, тем более чутко проявляется реагирование вашего приложения.

Не ждите долгой процедуры завершения, прежде чем выводить результаты пользователям. Если нет отображения на экране дисплея, пока не появятся все результаты, пользователи могут интерпретировать наблюдаемое как инерционность вашей программы. Вместо этого нужно выводить на дисплей частичные результаты как можно скорее и после этого позволить инициирование долгой процедуры, так что у пользователей появится что-то полезное, на что можно посмотреть, в то время как остальная задача завершается.

Используйте индикатор прогресса, чтобы помочь пользователям оценить, как много времени займет процесс завершения работы. Пользователям не всегда нужно точно знать, сколько времени займет задача, но важно дать его оценку. Например, при необходимости вы можете использовать индикатор в сочетании с информативным текстом, таким как «Осталось две минуты» или «Копирование 30 из 850 файлов».

6.1.15. Предоставление альтернативных способов выполнения задач

Разные люди могут взаимодействовать с приложениями по-разному, в зависимости от личных предпочтений и обстоятельств. Так как вы не можете предсказать, как именно пользователи будут использовать ваше приложение, нужно предложить несколько способов для выполнения каждой учебной задачи.

6.1.16. Поддержка доступности команд приложения

Сделайте приложение команд доступным в меню. Строка меню является первым местом, где пользователи склонны искать команды, особенно когда они еще не знакомы с приложением. Хорошая идея – разместить списки всех важных команд в соответствующем меню приложения, чтобы пользователи могли их легко найти. Обеспечьте клавиатуру только в качестве альтернативы. Многие люди предпочитают использовать клавиатуру лишь совместно с мышью или трекпадом.

Есть два основных пути оказания поддержки клавиатуры пользователем:

1. Соблюдение стандартных сочетаний клавиш и создание приложений с конкретными ярлыками для часто используемых команд.

2. Добавление поддержки режима полного доступа с клавиатуры на все элементы пользовательского интерфейса. Режим полного доступа к клавиатуре позволяет пользователям перемещаться и активировать окна, меню, элементы интерфейса и системные возможности, используя только клавиатуру.

6.1.17. Правильное использование стандартных элементов интерфейса

Windows предоставляет широкий спектр встроенных элементов пользовательского интерфейса. Это — элементы управления, меню и диалоговые окна. Использование их в вашем приложении дает несколько важных преимуществ:

- пользователи знакомы с встроенными элементами, так что они уже знают, как использовать их в вашем приложении;
- сокращается время разработки, потому что вы не должны разрабатывать полный набор пользовательских элементов;
- предоставляемые операционной системой элементы помогают обеспечить согласованность вашего приложения с остальной частью системы;
- если есть изменения, предоставляемые системой управления, ваше приложение автоматически получает обновленный внешний вид.

Для реализации этих преимуществ крайне важно, чтобы вы правильно использовали встроенный элемент. Не назначайте новых моделей поведения для встроенных элементов пользовательского интерфейса. Важно использовать предоставляемые системой элементы строго в соответствии с изложенными в документах стандартами. Нельзя создавать пользовательский элемент интерфейса, копирующий поведение встроенных элементов.

6.1.18. Принципы оказания помощи пользователю

Помощь пользователю должна быть продуктивной и немедленной. Пользователи ожидают извлечь выгоду из вашего интерфейса и приложения, как только они установлены.

Обеспечьте их ожидание, следуя следующим инструкциям:

1. Не спрашивайте пользователей о разрешении на перезагрузку системы. После установки приложения должны быть готовы к немедленному использованию. Если ваше приложение для корректной работы требует перезагрузку, то вы должны заново продумать его дизайн.

2. Избегайте требований от пользователей предоставить множество сведений об установке, прежде чем они могут сделать что-нибудь еще.

Часто вместо того чтобы пользователи могли сразу приступить к работе, их просят сделать установки и настройки. Делайте это только тогда, когда это действительно необходимо. Создайте интеллектуальные настройки по умолчанию. Определите, как большинство пользователей будет, скорее всего, использовать ваше приложение, и установите соответствующие настройки по умолчанию. Когда вы точно оцените потребности вашей аудитории пользователей, то потребность в настройках отпадет.

3. Убедитесь, что функциональность органов управления в вашем приложении легко обнаружить.

6.1.19. Применение анимации

Анимация может быть одним из лучших способов для обеспечения обратной связи и уточнения результатов действий пользователя. Как правило, анимация также является одной из самых легких частей пользовательского интерфейса, где проектировщики постоянно ошибаются.

Используйте следующие рекомендации, чтобы повысить качество вашего приложения за счет полезных анимаций.

Избегайте бесполезных анимаций. Бесцельная и нелогичная анимация быстро становится утомительной и раздражает пользователей. Будьте уверены, что вводимая вами анимация улучшает понимание пользователем функциональности вашего приложения.

Старайтесь не использовать анимацию как центр внимания пользователя. Если слишком много внимания уделяется анимации в вашем приложении, пользователи, вероятно, станут отвлекаться от своей основной задачи. Лучше, если анимация помогает пользователям понять, что происходит, не привлекая к себе особого внимания.

Использование анимации позволяет прояснить последствия действий пользователя. Анимация переходов в окна из полного экрана — это хорошая идея, чтобы создать гладкую, высококачественную анимацию для замены стандартного перехода.

Использование анимации для повышения реализма. Если вы создаете пользовательские элементы, которые выглядят, как реальные физические объекты, не забудьте оживить их движения таким образом, чтобы они были также реальными.

Использование анимации в процессе изменения свойств объекта. Демонстрация переходов объекта из одного состояния в другое, вместо того чтобы показывать только начальное и конечное состояние, помогает пользователям понять, что происходит, и усиливает чувство контроля над

процессом. Используйте анимацию, когда действие происходит так быстро, что пользователи не могут отслеживать его.

Избегайте анимирования всего. Хотя и возникает мысль, что увеличение анимации приводит к лучшей обратной связи, но это, вообще говоря, неверно. Большинство задач и действий в приложении лучше всего выполнять быстро и с минимумом украшений.

Избегайте анимации рутинных действий, предоставляемых системой управления. Пользователи понимают, как работают общие элементы пользовательского интерфейса, и они не ценят, когда их вынуждают тратить дополнительное время на просмотр ненужных анимаций всякий раз, когда они нажимают кнопки или переключатели вкладок.

6.1.20. Обеспечение безопасного исследования приложения пользователем

Поощряйте пользователей к изучению вашего приложения путем прощения ошибок, т.е. делая большинство действий с интерфейсом легко обратимыми. Пользователи должны чувствовать, что они могут попытаться сделать что-либо без повреждения системы, не ставя под угрозу данные. В идеале ваше приложение предоставляет пользователям возможности, которые им необходимы, помогая им избежать опасных, необратимых действий. Пользователи должны чувствовать себя комфортно, используя ваше приложение. Кроме того, нужно рассмотреть вопрос о предоставлении пользователям возможности вносить изменения в содержание, но отложить совершение происходящих при этом изменений на более позднее время. Необходимо предупреждать пользователей, когда они иницируют задачу, которая будет вызывать неожиданные и необратимые потери данных. Необходимо обеспечить большое количество информативной обратной связи и взаимодействия на всем протяжении времени работы вашего приложения. Соответствующая надежная обратная связь помогает пользователям быть уверенными, что у них есть достаточно информации, чтобы сделать правильный выбор.

Насколько возможно, нужно дать пользователям возможность делать то, что они хотят, без лишних помех. Завоевать доверие пользователей позволяют надежность, предсказуемость и стабильность приложения. Надежное стабильное приложение ведет себя предсказуемым образом и поддерживает целостность данных пользователя, делая все возможное для предотвращения потери или повреждения данных. Убедитесь, что элементы пользовательского интерфейса ведут себя так, как пользователи ожидают от них. Когда элементы пользовательского интерфейса работают, как ожидается, пользователи больше доверяют вашему приложению. В частности, крайне важно избежать изменения смысла стандартных элементов пользовательского интерфейса, знакомых пользователю.

Предоставляйте пользователю информацию о состоянии системы и обратной связи, чтобы показать, что ваше приложение выполняет указанные задачи. Пользователи должны верить, что ваше приложение действует по их командам. Убедитесь, что пользователи будут получать предсказуемый результат из его содержания. Если ваше приложение позволяет пользователям форматировать данные для печати, публикации в сети Интернет или записывать на пленку, *DVD* или носители других форматов, убедитесь, что нет никаких существенных различий между тем, что пользователи видят на экране, и тем, что они получают в конечной продукции. Ваша задача — предвидеть ошибки и помочь пользователям избежать их, когда это возможно.

6.1.21. Обеспечение адаптации пользователя к изменениям окружающей среды

Адаптируемая программа — та, которая регулирует надлежащим образом свою конфигурацию, т.е. она не перестает работать, когда изменяются текущие внешние условия. Прежде всего будьте готовы к изменениям конфигурации сети и убедитесь, что пользователи могут извлечь выгоду из вашего приложения, если их скорость подключения к сети незначительна. Во-вторых, нужно быть консервативным с энергопотреблением вашего приложения, так чтобы мобильные пользователи могли использовать свой компьютер перед зарядкой как можно дольше.

Нужно предвидеть изменения размера дисплея, его разрешение. Убедитесь, что ваша графика хорошо выглядит на всех разрешениях экрана, и будьте готовы настроить окно местоположения и размеров. Будьте готовы обеспечить для разных пользователей различные привилегии.

Функция быстрого переключения пользователей позволяет нескольким пользователям войти в систему одновременно и быстро переключаться между учетными записями. Чтобы убедиться, что ваше приложение ведет себя соответствующим образом, когда есть несколько пользователей, которые вошли в систему, или если текущий пользователь имеет ограниченные привилегии, соблюдайте следующее.

Некоторые пользователи могут работать под ограниченными правами и имеют ограниченный доступ к некоторым частям системы. Пользователи на компьютере могут включать в себя как местных, так и пользователей сети, так что не думайте, что домашний каталог пользователя находится на локальном томе. Будьте готовы к тому, что вы обращаетесь к сети. В именованные ресурсы, которые могут быть потенциально доступны для приложений из нескольких сессий, пользователь должен включать

идентификатор сессии в имя ресурса. Это относится к кэш-файлам, разделяемой памяти, семафорам и именованным каналам.

Избегайте зависимости от наличия конкретных аппаратных средств. Аппаратная конфигурация может сильно варьировать в зависимости от компьютера, страны и пользователя. Избегайте делать предположения, основанные на локальных данных текущего пользователя. Старайтесь проверить работоспособность приложения у других разработчиков с другими пользовательскими установками.

6.1.22. Проектирование взаимодействия приложений

Совместимые приложения легко общаются с другими приложениями и услугами, так что пользователи могут открывать их содержание в различных приложениях и обмениваться информацией с другими пользователями. Пользователи оценят совместимость, поскольку это означает, что они могут сосредоточиться на содержании и не обращать внимания на детали, такие как форматы файлов и протоколы обмена данными. Насколько возможно, старайтесь не использовать пользовательские форматы файлов. Вместо этого следует использовать стандартные форматы файлов, чтобы пользователи могли легко обмениваться документами с другими пользователями или открывать их в различных приложениях. Если вы должны использовать собственный формат файла, убедитесь, что обеспечиваются возможности импорта и экспорта, так что пользователи могут обмениваться данными с другими приложениями и системой. Используйте тот же формат файлов на всех платформах, которые вы поддерживаете. Это гарантирует, что пользователи могут использовать ваше приложение, чтобы просмотреть его содержание независимо от устройства или используемой платформы.

6.1.23. Поддержка расширений файлов

Расширение файла определяет тип документа таким образом, чтобы все платформы могли его понять. Использование стандартных протоколов для обмена данными *.XML* является предпочтительным форматом для обмена данными между приложениями и платформами, потому что он — кросс-платформенный и широко поддерживается разработчиками.

6.1.24. Обеспечение доступной пользователю помощи

В идеале пользователи могут легко понять, как использовать ваше приложение, даже не читая руководство, но иногда они нуждаются в небольшой помощи для понимания, как использовать расширенные функции или как изменить главную задачу. Предоставление помощи осуществляется с помощью тегов, которые описывают, как использовать элементы пользовательского интерфейса. Помощь состоит из небольших

тегов, в которых отображаются краткие описания того, что делает элемент пользовательского интерфейса. Лучше всего, когда использование пользовательского интерфейса вашего приложения становится мгновенно очевидным для пользователей. Метки могут быть хорошим способом помощи начинающим пользователям и не раздражают опытных пользователей.

Предоставление помощи может быть в виде книги (справки), которая описывает, как выполняются задачи в вашем приложении. Пользователи открывают книгу помощи приложения, когда им не понятно, как достичь своей цели. Хотя пользователи могут обратиться к книге помощи, чтобы узнать, как использовать определенный элемент управления, но они с большей вероятностью будут обращаться за помощью при решении задач более высокого уровня. По этой причине книга помощи должна быть сформирована на основе задач и по большей части, должна описать использование приложения в контексте выполнения задачи.

6.1.25. Обеспечение удобного ввода данных

Есть несколько способов, с помощью которых пользователям, возможно, понадобится вводить информацию в приложения, такую как заполнить форму, ввести пароль. По возможности (и необходимости) автоматически заполняйте текстовые поля информацией о пользователях. Пользователи оценят удобство получаемой информации при условии, если она верна. Если вы не можете гарантировать точность информации, которую вы предоставили, то лучше оставить текстовые поля пустыми. Если вы предоставляете текст, не забудьте указать поля, которые вы заполняете (возможно, выделив их), так чтобы пользователь мог легко отличать информацию, обеспечиваемую вашим приложением, от информации, которую предоставляют они сами. Обратите внимание, что поле пароля всегда должно быть пустым.

Выполните соответствующие проверки того, как пользователи вводят информацию. Например, если единственным разрешенным значением для поля является строка из цифр, то приложение должно предупредить пользователей, как только они введут нецифровой символ. Проверка вводимых пользователем данных помогает избежать ошибок, вызванных неожиданными данными.

6.1.26. Использование терминологии, ориентированной на пользователей

Почти все приложения должны использовать какие-то слова, чтобы общаться с пользователями, даже если это — только слова в надписях на кнопках. Важно выбрать все слова вашего приложения столь тщательно,

чтобы можно было гарантировать, что ваше общение с пользователями однозначно и точно. В общем случае избегайте жаргона.

Используйте терминологию, которая устраивает пользователей. Если ваше приложение — для искушенных пользователей, которые часто используют набор специализированных технических терминов, то имеет смысл использовать эти термины. С другой стороны, если ваша пользовательская аудитория состоит в основном из начинающих или случайных пользователей, лучше использовать более простые, обычно понимаемые термины.

Избегайте условностей, используемых при разработке. Как разработчик вы ссылаетесь на элементы пользовательского интерфейса приложения и процессы непонятным для большинства пользователей образом. Не забудьте внимательно изучить пользовательский интерфейс для замены терминов, применяемых разработчиками, на термины, соответствующие условиям пользователя.

6.2. Общие правила дизайна при создании мультимедийных интерфейсов

Настоящие рекомендации предназначены для разработчиков мультимедийных интерфейсов, не имеющих специального образования в области дизайна, технической эстетики, педагогики и эргономики. Их назначение — изложение основных принципов дизайна мультимедийных интерфейсов для исключения грубых ошибок проектирования.

6.2.1. Правила шрифтового оформления

1. Шрифты с засечками читаются легче, чем гротески (шрифты без засечек).
2. Для основного текста не рекомендуется использовать прописные буквы.
3. Шрифтовой контраст можно создать посредством размера шрифта, его толщины, начертания, формы, направления и цвета.

6.2.2. Правила выбора цветовой гаммы

1. Цветовая гамма должна состоять не более чем из двух-трех цветов.
2. Существуют несочетаемые комбинации цветов.
3. Черный цвет имеет негативный (мрачный) подтекст.
4. Белый текст на черном фоне читается плохо (инверсия плохо читается).

6.2.3. Правила общей композиции

1. На экране не должно быть больше шести значимых объектов, так как человек не в состоянии оперировать за один раз более чем с шестью объектами и пунктами чего-либо.
2. Логотип разработчика интерфейса на экране должен располагаться справа внизу (слева наверху и т. д.).
3. Логотип должен быть простой и лаконичной формы.
4. Дизайн должен быть простым, а текст — коротким.
5. Крупные объекты в составе любой композиции смотрятся довольно нелепо. Аршинные буквы в заголовках, кнопки навигации высотой в 40 пикселей, верстка в одну колонку шириной в 600 точек, разделитель одного цвета, растянутый на весь экран, — все это придает дизайну интерфейса непрофессиональный вид, и этого следует избегать.

6.2.4. Общие стилевое оформление

1. Стил может включать определенный шрифт (гарнитура и цвет), цвет фона или фоновый рисунок, декоративный элемент небольшого размера и др.
2. Оформление интерфейса не должно отвлекать внимание пользователя от его содержательной части.
3. Все окна программного интерфейса должны быть выдержаны в одном стиле.

6.2.5. Рекомендации по дизайну мультимедийного представления информации

Чтобы мультимедийная информация интерфейса хорошо воспринималась пользователем и не вызывала отрицательных эмоций (подсознательных или осознанных), необходимо соблюдать правила ее оформления.

Программный интерфейс предполагает сочетание информации различных типов (текста, графических изображений, музыкальных и звуковых эффектов, анимации и видеофрагментов), поэтому необходимо учитывать специфику комбинирования фрагментов информации. Кроме того, оформление и демонстрация каждого из перечисленных типов информации также подчиняются определенным правилам. Например, для текстовой информации важен выбор шрифта, для графической — яркости и насыщенности цвета, для наилучшего их совместного восприятия необходимо оптимальное взаиморасположение на рабочем поле. Рассмотрим рекомендации по оформлению и представлению на экране материалов различного вида.

6.2.6. Текстовая информация

1. Размер шрифта: 24–54 пункта (заголовок), 18–36 пунктов (обычный текст).
2. Цвет шрифта и цвет фона должны контрастировать (текст должен хорошо читаться), но не резать глаза.
3. Тип шрифта: для основного текста гладкий шрифт без засечек (*Arial, Tahoma, Verdana*); для заголовка можно использовать декоративный шрифт, если он хорошо читаем.
4. Курсив, подчеркивание, жирный шрифт, прописные буквы рекомендуется использовать только для смыслового выделения фрагментов текста.

6.2.7. Графическая информация

1. Рисунки, фотографии, диаграммы призваны дополнить текстовую информацию или передать ее в более наглядном виде.
2. Желательно избегать рисунков, не несущих смысловой нагрузки, если они не являются частью стилевого оформления.
3. Цвет графических изображений не должен резко контрастировать с общим стилевым оформлением интерфейса.
4. Иллюстрации рекомендуется сопровождать пояснительным текстом.
5. Если графическое изображение используется в качестве фона, то текст на этом фоне должен быть хорошо читаем.

6.2.8. Эффекты анимации

Анимационные эффекты используются для привлечения внимания пользователя или для демонстрации динамики развития какого-либо процесса. В этих случаях использование анимации оправдано, но не стоит чрезмерно насыщать интерфейс такими эффектами, иначе это вызовет негативную реакцию и быстрое утомление пользователя.

6.2.9. Использование звука

1. Звуковое сопровождение должно отражать суть или подчеркивать особенность интерфейса системы голосовых подсказок.
2. Необходимо выбрать оптимальную громкость, чтобы звук был слышен, но не оглушителен.
3. Если используется фоновая музыка, то она должна не отвлекать внимание пользователя и не заглушать слова диктора.
4. При групповой деятельности необходимо обеспечить защиту пользователя от звуковой информации с соседних рабочих мест.

Чтобы все материалы интерфейса воспринимались целостно и не возникало диссонанса между отдельными его фрагментами, необходимо учитывать приведенные ниже общие правила оформления.

6.2.10. Единое стилевое оформление

1. Стилль может включать определенный шрифт (гарнитура и цвет), цвет фона или фоновый рисунок, декоративный элемент небольшого размера и др.
2. Не рекомендуется использовать в стилевом оформлении интерфейса более 3 цветов и более 3 типов шрифта.
3. Оформление слайда рабочего поля интерфейса не должно отвлекать внимание пользователей от его содержательной части.
4. Все рабочие экраны интерфейса должны быть выдержаны в одном стиле.

6.2.11. Содержание и расположение информационных блоков

1. Информационных блоков не должно быть слишком много (3–6).
2. Рекомендуемый размер одного информационного блока – не более 1/2 размера рабочего поля.
3. Желательно присутствие на странице блоков с разнотипной информацией (текст, графики, диаграммы, таблицы, рисунки), дополняющей друг друга.
4. Ключевые слова в информационном блоке необходимо выделить.
5. Информационные блоки лучше располагать горизонтально, связанные по смыслу блоки – слева направо.
6. Наиболее важную информацию следует поместить в центр поля зрения оператора.
7. Логика предъявления информации на рабочем поле должна соответствовать логике алгоритма работы оператора.

6.2.12. Общие правила оформления текста

Помимо правильного расположения текстовых блоков, нужно не забывать и об их содержании — тексте. В нем ни в коем случае не должно содержаться орфографических ошибок.

1. Точка в конце заголовка и подзаголовков, выключенных отдельной строкой, не ставится. Если заголовок состоит из нескольких предложений, то точка не ставится после последнего из них. Порядковый номер всех видов заголовков, набираемый в одной строке с текстом, должен быть отделен пробелом независимо от того, есть ли после номера точка.

2. Точка не ставится в конце подрисуночной подписи, в заголовке таблицы и внутри нее. При отделении десятичных долей от целых чисел надо ставить точку (0.158).

3. Перед знаком препинания пробел не ставится (исключение составляют открывающиеся парные знаки, например скобки, кавычки). После знака препинания пробел обязателен (если этот знак не стоит в конце абзаца). Тире выделяется пробелами с двух сторон. Дефис пробелами не выделяется.

4. Числительные порядковые и количественные выражаются в простом тексте словами (обычно однозначные при наличии сокращенных наименований), цифрами (многозначные и при наличии сокращенных обозначений) и смешанным способом (после десятков тысяч часто применяются выражения типа 25 тыс.), числительные в косвенных падежах набирают с так называемыми «наращениями» (6-го). В наборе встречаются арабские и римские цифры.

5. Индексы и показатели между собой и от предшествующих и последующих элементов набора не должны быть разделены пробелом (H₂O, м³/с).

6. Нельзя набирать в разных строках фамилии и инициалы, к ним относящиеся; один инициал от другого нужно отделять пробелом.

7. Не следует оставлять в конце строки предлоги и союзы (из одной-трех букв), начинающие предложение, а также однобуквенные союзы и предлоги в середине предложений.

8. Последняя строка в абзаце не должна быть слишком короткой. Надо стараться избегать оставления в строке или переноса двух букв. Текст концевой строки должен быть в 1,5–2 раза больше размера абзацного отступа, т. е. содержать не менее 5–7 букв. Если этого не получается, необходимо вогнать остаток текста в предыдущие строки или выгнать из них часть текста. Это правило не относится к концевым строкам в математических рассуждениях, когда текст может быть совсем коротким, например «и», «или» и т. п.

9. Знаки процента (%) применяют только с относящимися к ним числами, от которых они отделяются неразрывным пробелом.

10. Знаки градуса (°), минуты (′), секунды (″) от предыдущих чисел не должны быть отделены пробелом, а от последующих чисел должны быть отделены пробелом (10° 15′).

11. Формулы в текстовых строках набора научно-технических текстов должны быть отделены от текста на пробел или на двойной пробел. Формулы, следующие в текстовой строке одна за другой, должны быть отделены друг от друга удвоенными пробелами.

12. Знаки номера (№) и параграфа (§) применяют только с относящимися к ним числами и отделяются пробелом от них и от остального текста с двух сторон. Сдвоенные знаки набирают вплотную друг к другу. Если к знаку относится несколько чисел, то между собой они отделяются пробелами. Нельзя в разных строках набирать знаки и относящиеся к ним цифры.

13. В русском языке различают следующие виды сокращений: буквенная аббревиатура — сокращенное слово, составленное из первых букв слов, входящих в полное название (СССР, НДР, РФ, вуз); сложносокращенные слова, составленные из частей сокращенных слов (колхоз) или усеченных и полных слов (Моссовет), и графические сокращения по начальным буквам (г. — год), по частям слов (см. — смотри), по характерным буквам (млрд — миллиард), а также по начальным и конечным буквам (ф-ка — фабрика). Кроме того, в текстах применяют буквенные обозначения единиц физических величин. Все буквенные аббревиатуры набирают прямым шрифтом без точек и без разбивки между буквами, сложносокращенные слова и графические сокращения набирают как обычный текст. В выделенных шрифтами текстах все эти сокращения набирают тем же выделительным шрифтом.

6.2.13. Специфические требования при компьютерном наборе текста

1. При наборе текста одного абзаца клавиша «Перевод строки» («Enter») нажимается только в конце этого абзаца.

2. Между словами нужно ставить ровно один пробел. Равномерное распределение слов в строке текстовым процессором выполняется автоматически. Абзацный отступ (красную строку) устанавливать с помощью пробелов запрещено; для этого используются возможности текстового процессора (например, можно использовать бегунки на горизонтальной полосе прокрутки или табулятор).

Знак неразрывный пробел (Вставка → Символ, вкладка Специальные знаки или комбинация клавиш *CTRL+SHIFT*+пробел) препятствует символам, между которыми он поставлен, располагаться на разных строчках и сохраняется фиксированным при любом выравнивании абзаца (не может увеличиваться в отличие от обычного пробела).

3. Выделением называют особое оформление отдельных слов или частей текста, которое подчеркивает их значение. Все виды выделений делят на три группы:

- шрифтовые выделения, выполняемые путем замены характера или начертания шрифта, — набор курсивом, полужирным, жирным,

полужирным курсивом, прописными или капительными буквами, шрифтами другого кегля или даже другой гарнитуры;

- не шрифтовые выделения, выполняемые путем изменения расстояний между буквами (набор вразрядку) или между строками набора (дополнительные отбивки отдельных строк), изменения формата набора (набор «в красную строку», набор с одно- или двусторонними втяжками), подчеркивания текста тонкими или полужирными линейками или заключения отдельных частей текста в рамки;
- комбинированные выделения, выполняемые одновременно двумя способами; например, набор полужирным вразрядку, набор полужирным шрифтом увеличенного кегля с выключкой в «красную строку» и дополнительными отбивками, набор курсивом с заключением текста в рамку и т. п.

4. Шрифтовые выделения (курсивом, полужирным, жирным) должны быть выполнены шрифтами той же гарнитуры и кегля, что и основной текст. Знаки препинания, следующие за выделенной частью текста, должны быть набраны шрифтом основного текста.

5. Знак тире, или длинное тире, может быть набран с помощью одновременного нажатия комбинации клавиш *CTRL+SHIFT*+серый минус или Вставка → Символ, вкладка Специал.

6.3. Контрольные вопросы

1. Что такое «философия проектирования интерфейса»?
2. Назовите характеристики, поддерживающие ментальную модель пользователя.
3. Сколько цветов и типов шрифта целесообразно использовать при оформлении графического интерфейса?
4. Как поставить в тексте неразрывный пробел?
5. Перечислите общие правила оформления текста. Приведите примеры к каждому пункту правил.
6. В каких случаях при проектировании пользовательского интерфейса целесообразно использование графической информации?
7. Приведите правила оформления мультимедийных интерфейсов и проиллюстрируйте их применение.
8. Какие преимущества дает использование стандартных элементов интерфейса?

9. Приведите примеры явных и подразумеваемых действий, реализуемых в интерфейсе вашего телефона.

10. Опишите приемы, которые вы можете применить для обеспечения высокого эстетического качества интерфейса.

6.4. Темы для дискуссии

1. Проведите эргономическую экспертизу рабочих мест операторов энергетической системы на предмет выполнения в ней пунктов эргономических требований ГОСТ Р ИСО 6385-2007, касающихся распределения функций.
2. Какими эргономическими принципами обеспечивается безопасность интерфейсов оборудования (обсуждение ГОСТ Р ЕН 614-1-2003)?
3. Разработайте проект дизайна интернет-сайта для оформления курортных путевок с учетом требований юзабилити.

Рекомендуемая литература

1. *Баканов А. С.* Эргономика пользовательского интерфейса: от проектирования к моделированию человеко-компьютерного взаимодействия / А. С. Баканов, А. А. Обознов. — Институт психологии РАН, 2011. — 176 с.
2. *Баканова Н. Б.* Разработка интерфейсов пользователя корпоративных информационных систем / Н. Б. Баканова, А. А. Обознов. — Тверь: Триада, 2008. — 112 с.
3. *Зинченко Т. П.* Опознание и кодирование / Т. П. Зинченко. — Л.: Изд-во ЛГУ, 1981. — 183 с.
4. *Калиновский А. И.* Юзабилити: как сделать сайт удобным / А. И. Калиновский. — Минск: Новое знание, 2005. — 220 с.
5. *Кирсанов Д.* Веб-дизайн: книга Дмитрия Кирсанова / Д. Кирсанов. — СПб: Символ-Плюс, 2001. — 376 с.
6. *Кориков А. М.* Система «Человек — компьютер»: на пути создания человеко-ориентированного интерфейса / А. М. Кориков, О. А. Кривцов. — Томск: В-Спектр, 2010. — 183 с.
7. *Купер А.* Алан Купер об интерфейсе. Основы проектирования взаимодействия / А. Купер, Р. Рейман, Д. Кронин. — СПб: Символ-Плюс, 2009. — 688 с.
8. *Львов В. М.* Эргономика. Вводный курс. Учебное пособие для вузов / В. М. Львов, П. Я. Шлаен. — Тверь: Триада, 2004. — 188 с.
9. *Магазанник В. Д.* Человеко-компьютерное взаимодействие: учебное пособие для вузов / В. Д. Магазанник, В. М. Львов. — Тверь: Триада, 2005. — 200 с.
10. *Нильсен Я.* Веб-дизайн. Книга Якоба Нильсена / Я. Нильсен. — СПб: Символ-Плюс, 2006. — 512 с.
11. *Нильсен Я.* Web-дизайн: удобство использования Web-сайтов / Я. Нильсен, Х. Лоранжер. — М.: Вильямс, 2007. — 366 с.
12. *Нильсен Я.* Веб-дизайн: анализ удобства использования веб-сайтов по движению глаз / Я. Нильсен, К. Перниче. — М.: Вильямс, 2010. — 480 с.
13. *Норман Д.* Дизайн привычных вещей / Д. Норман. — М.: Вильямс, 2006. — 384 с.
14. *Раскин Д.* Интерфейс: новые направления в проектировании компьютерных систем / Д. Раскин. — СПб: Символ-плюс, 2007. — 272 с.
15. *Сергеев С. Ф.* Инженерная психология и эргономика: учебное пособие / С. Ф. Сергеев. — М.: НИИ школьных технологий, 2008. — 176 с.
16. *Сергеев С. Ф.* Введение в инженерную психологию и эргономику иммерсивных сред: учеб. пособие / С. Ф. Сергеев. — СПб: СПбГУ ИТМО, 2011. — 258 с.
17. *Сергеев С. Ф.* Введение в проектирование интеллектуальных интерфейсов: учебное пособие / С. Ф. Сергеев, П. И. Падерно, Н. А. Назаренко. — СПб: СПбГУ ИТМО, 2011. — 108 с.
18. *Сергеев С. Ф.* Юзабилити-тестирование интерфейсов информационных систем в гуманитарных науках и искусстве: учебное пособие / С. Ф. Сергеев. — СПб: Изд-во С.-Петерб. ун-та, 2012. — 86 с.
19. *Сергеев С. Ф.* Методы юзабилити в тренажерах и обучающих системах / С. Ф. Сергеев, В. Н. Соколов, Г. Л. Коротеев. — СПб: Изд-во Политехн. ун-та, 2012. — 132 с.
20. *Унгер Р.* UX-дизайн. Практическое руководство по проектированию опыта взаимодействия / Р. Унгер, К. Чендлер. — СПб: Символ-Плюс, 2011. — 336 с.
21. *Шлаен П. Я.* Эргономика для инженеров. Эргономическое обеспечение проектирования человеко-машинных комплексов: проблемы, методология, технологии / П. Я. Шлаен, В. М. Львов. — Тверь: ТвГУ, 2004. — 476 с.

Приложение 1

Список стандартов по проектированию пользовательских интерфейсов и юзабилити

1. ЕН 614-1:1995. Безопасность машин. Принципы эргономического проектирования. Часть 1. Терминология и основные принципы (EN 614-1:1995) (Safety of machinery - Ergonomic design principles — Part 1: Terminology and general principles).
2. ЕН 614-2: 2000. Безопасность машин. Принципы эргономического проектирования. Часть 2. Взаимосвязь между производственным оборудованием и производственными задачами. (EN 614-2:2000) (Safety of machinery — Ergonomic design principles — Part 2: Interactions between the design of machinery and work tasks).
3. ИСО/ТР 16982:2002. Эргономика взаимодействия «человек — система». Методы, основанные на удобстве применения для обеспечения проектирования, ориентированного на человека. (ISO/TR 16982:2002) (Ergonomics of human-system interaction — Usability methods supporting human-centred design).
4. ИСО 13407:1999. Процессы человекоориентированного проектирования для интерактивных систем. (ISO 13407:1999) (Human-centred design processes for interactive systems).
5. ИСО 9241-2:1992. Эргономические требования для офисных работ с видеодисплейными терминалами (VDTs): Часть 2. Требования к постановке задач. (ISO 9241-2:1992) (Ergonomic requirements for office work with visual display terminal (VDTs) — Part 2: Guidance on task requirements).
6. ИСО 10075:1991. Эргономические принципы, относящиеся к умственной нагрузке. Основные термины и определения (ISO 10075:1991) (Ergonomic principles related to mental work-load — General terms and definitions).
7. ИСО 9241-11. Эргономические требования, связанные с использованием видеотерминалов для учрежденческих работ. Часть 11. Руководство по определению и измерению используемости. (ISO 9241-11) (Ergonomic requirements for office work with visual display terminal (VDTs) — Part 11: Guidance on usability).
8. ГОСТ 28806-90. Качество программных средств. Термины и определения. (Software quality. Terms and definitions). М.: Изд-во стандартов, 1992.
9. ГОСТ Р ИСО/МЭК 15910-2002. Информационная технология. Процесс создания документации пользователя программного средства. М.: Изд-во стандартов, 2002.
10. ГОСТ Р 6385-2007. Эргономика. Применение эргономических принципов проектировании производственных систем. М.: Стандартинформ, 2008.
11. ГОСТ Р ИСО 14915-1-2010. Эргономика мультимедийных пользовательских интерфейсов. Часть 1. Принципы проектирования и структура. Ergonomics of multimedia user interfaces. Part 1. Design principles and framework. М.: Стандартинформ, 2011.
12. ГОСТ Р ИСО 9241-110-2009. Эргономика взаимодействия «человек — система». Часть 110. Принципы организации диалога. Ergonomics of human-system interaction. Part 110. Dialogue principles. М.: Стандартинформ, 2010.
13. ГОСТ Р ИСО 9355-2-2009. Эргономические требования к проектированию дисплеев и механизмов управления. Часть 2. Дисплеи. М.: Стандартинформ, 2010.
14. ГОСТ Р ИСО 9241-2-2009. Эргономические требования к проведению офисных работ с использованием видеодисплейных терминалов (VDT). Часть 2. Требования к производственному заданию. М.: Стандартинформ, 2010.
15. ГОСТ Р ИСО 9355-1-2009. Эргономические требования к проектированию дисплеев и механизмов управления. Часть 1. Взаимодействие с человеком. М.: Стандартинформ, 2010.
16. ГОСТ Р ИСО 9241-5-2009. Эргономические требования к проведению офисных работ с использованием видеодисплейных терминалов (VDT). Часть 5. Требования к расположению рабочей станции и осанке оператора. М.: Стандартинформ, 2010.
17. ГОСТ Р ИСО 15537-2009. Эргономика. Принципы отбора испытателей для проверки антропометрических свойств промышленной продукции и конструкций. Ergonomics. Principles for selecting test persons for testing anthropometric aspects of industrial products and designs. М.: Стандартинформ, 2010.
18. ГОСТ Р МЭК 60073-2000. Интерфейс человеко-машинный. Маркировка и обозначения органов управления и контрольных устройств. Правила кодирования информации. М.: Стандартинформ, 2000.

Приложение 2

Список интернет-ресурсов по юзабилити

1. <http://www.usabilityfirst.com/> — *Usability First* — руководство по юзабилити.
2. <http://www.usability.ru/> — *Usability в России* — сайт предназначен для аккумуляции информационных ресурсов, связанных с эргономической теорией и практикой.
3. <http://www.usetheics.ru/> — *Usetheics* — дизайн интерфейсов и юзабилити-тестирование.
4. <http://www.webmasterpro.com.ua/forum/> — *WebmasterPro* — форум о поисковых системах. Обсуждение вопросов оптимизации сайтов для поисковых систем. Разбор конкретных сайтов на предмет оптимизации и юзабилити.
5. <http://webword.com/> — *WebWord* — юзабилити и человеческий фактор в сети Интернет.
6. <http://www.anybrowser.com/> — *AnyBrowser* — повышение пользовательских свойств веб-сайтов.
7. <http://www.webpagesthatsuck.com/> — *Web Pages That Suck* — здесь можно научиться хорошему дизайну веб-страниц на примере страниц с плохим дизайном.
8. <http://www-106.ibm.com/developerworks/usability/> — *DeveloperWorks* раздел Usability — удобство использования — это специальная тема на *DeveloperWorks* Веб-сайте, — источнике ресурсов для разработчиков приложений для электронного бизнеса, электронной коммерции и предприятий.
9. <http://www.usableweb.com/> — *Usable Web* — сайт содержит коллекцию, состоящую из 1400 ссылок и сопроводительной информации о человеческом факторе; в нем рассматриваются вопросы о пользовательском интерфейсе и дизайне, характерные для *World Wide Web*.
10. <http://www.theusabilitycompany.com/> — *The Usability Company* — веб-аналитика, удобство использования, доступность и онлайн-исследования рынка в Великобритании.
11. <http://www.usablewebs.com/> — *Usable Webs* — консалтинг для вашего сайта, веб-разработки руководящих принципов, содержания и развития сайта.

12. <http://www.octagoninternet.co.uk/> — *Octagon Internet* — Лондон-Интернет на основе решения компании предоставляют веб-дизайн сайта, разработку, эксплуатацию и совершенствование, вопросы поисковой оптимизации и позиционирования веб-сайта, юзабилити-консалтинг.
13. <http://www.supersites.com/> — *SuperSites* — веб-сайт юзабилити-специалистов. Высокоспециализированные консультанты помогут увеличить ваше присутствие в Интернете.
14. <http://www.usability.uk.com/> — *Usability by Design* — увеличить ваши продажи и увеличить удовлетворенность клиентов за счет улучшения юзабилити вашего веб-сайта или программного обеспечения.
15. <http://www.abacuswebusability.com/> — *Abacus Web Usability* — веб-сайт оценки удобства для пользователя и анализ юзабилити-тестирования.
16. <http://www.usabilitylab.ru/> — юзабилитилаб — юзабилити, принципы, технологии, консультации.



В 2009 г. университет стал победителем многоэтапного конкурса, в результате которого определены 12 ведущих университетов России, которым присвоена категория «Национальный исследовательский университет». Министерством образования и науки Российской Федерации была утверждена Программа развития государственного образовательного учреждения высшего профессионального образования «Санкт-Петербургский государственный университет информационных технологий, механики и оптики» на 2009–2018 г.г.

КАФЕДРА УПРАВЛЕНИЯ ГОСУДАРСТВЕННЫМИ ИНФОРМАЦИОННЫМИ СИСТЕМАМИ

Кафедра УГИС создана в 2011 г. на магистерском корпоративном факультете НИУ ИТМО.

Обучение по магистерской программе «Управление государственными информационными системами» направлено на приобретение теоретических знаний и практических навыков в сфере создания и развития ИТ-систем для нужд государственной власти и местного самоуправления.

Практическая часть обучения проходит на базе Центра технологий электронного правительства НИУ ИТМО, Санкт-Петербургского информационно-аналитического центра и других партнерских структур под руководством опытных экспертов и представителей органов власти.

Сергей Фёдорович Сергеев

МЕТОДЫ ТЕСТИРОВАНИЯ И ОПТИМИЗАЦИИ ИНТЕРФЕЙСОВ ИНФОРМАЦИОННЫХ СИСТЕМ

Учебное пособие

В авторской редакции

Дизайн обложки

Вёрстка

Корректор

Редакционно-издательский отдел Санкт-Петербургского
национального исследовательского университета информационных
технологий, механики и оптики

Зав. РИО

Лицензия ИД № 00408 от 05.11.99

Подписано к печати

Заказ №

Тираж 50 экз.

Отпечатано на ризографе

С.Н. Ушаков

Е.Е. Нестерова

Т.А. Асанович

Н.Ф. Гусарова