

Министерство науки и высшего образования Российской Федерации

Калужский филиал
федерального государственного бюджетного образовательного
учреждения высшего образования
**«Московский государственный технический университет
имени Н.Э. Баумана
(национальный исследовательский университет)»**
(КФ МГТУ им. Н.Э. Баумана)

В.О. Федоров

**УПРАВЛЕНИЕ РАЗРАБОТКОЙ СИСТЕМ
АВТОМАТИЗИРОВАННОГО ПРОЕКТИРОВАНИЯ**

Конспект лекций

Калуга - 2019

Данный конспект лекций издается в соответствии с учебным планом кафедры «Системы обработки информации» для направления подготовки 09.03.01 «Информатика и вычислительная техника» КФ МГТУ им. Н.Э. Баумана.

Указания рассмотрены и одобрены:

- кафедрой «Системы обработки информации» (ИУ5-КФ)

протокол № 3 от 27 ноября 2019 г.

Зав кафедрой ИУ5-КФ  к.ф.-м.н., доцент Вершинин Е.В.

-методической комиссией факультета ИУ-КФ


протокол № 6 от 02 12 2019 г.

Председатель методической комиссии  к.т.н., доцент Адкин М.Ю.


-методической комиссией Калужского филиала

протокол № 4 от 24 12 2019 г.


Председатель методической комиссии

 д.э.н., профессор Перерва О.Л.

Рецензент:

 зав. кафедрой ИУ4-КФ, к.т.н. Гагарин Ю.Е.

Автор:

 к.т.н., доцент Федоров В.О.

Аннотация

Конспект лекций составлен в соответствии с утвержденной программой дисциплины «Управление разработкой систем автоматизированного проектирования», содержит теоретические сведения об основных функциях и требованиях пользовательских интерфейсов, этапах их проектирования, о прототипировании и тестировании интерфейсов.

© В.О. Федоров, 2019 г.

© Калужский филиал МГТУ им. Н.Э. Баумана, 2019 г.

ОГЛАВЛЕНИЕ

| | |
|--|----|
| ВВЕДЕНИЕ | 4 |
| Модуль 1 «Проектирование взаимодействия» | 5 |
| Лекция 1.1. Введение в целеориентированное проектирование. Ментальные модели | 5 |
| Лекция 1.2. Целевая аудитория, модели пользователей и их цели | 8 |
| Лекция 1.3. Сценарии поведения и использования | 11 |
| Модуль 2. «Элементы пользовательского интерфейса» | 18 |
| Лекция 2.1. Структура интерфейса..... | 18 |
| Лекция 2.2. Принципы визуального дизайна и идиомы в пользовательских интерфейсах..... | 24 |
| Лекция 2.3. Проектирование и доработка на примере стандартных элементов интерфейса..... | 30 |
| Список использованных источников. | 43 |

ВВЕДЕНИЕ

Настоящий конспект лекций составлен в соответствии с программой дисциплины «Управление разработкой систем автоматизированного проектирования» на кафедре «Системы обработки информации» факультета «Информатика и управление» Калужского филиала МГТУ им. Н.Э. Баумана.

Предназначен для студентов 4-го курса бакалавриата КФ МГТУ им. Н.Э. Баумана, обучающихся по направлению подготовки 09.03.01 «Информатика и вычислительная техника» и содержит теоретические сведения об основных функциях и требованиях пользовательских интерфейсов, этапах их проектирования, о прототипировании и тестировании интерфейсов, необходимые при освоении 1 и 2 модулей указанной выше дисциплины.

Модуль 1 «Проектирование взаимодействия»

Лекция 1.1. Введение в целеориентированное проектирование.

Ментальные модели

Маркетинговые исследования

Подобно опросам и фокус-группам маркетинговые исследования, могут стать ценным источником общей информации о пользователях. Эти методы особенно эффективны, если существует чёткая формулировка, того, какую информацию мы хотим получить с их помощью. Нужно ли нам знать, как ведут себя пользователи, когда взаимодействуют с какой-то конкретной функцией продукта? Или мы уже это 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.

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. Недостатком 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 его естественной среде обитания.

Метод карточной сортировки

Карточная сортировка — техника, снискавшая 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 web-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, может стать 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.

Анализ рабочих заданий

Анализ рабочих заданий — набор методов, использующих анкетирование или открытое интервью для формирования детального представления о том, как люди в настоящий момент выполняют конкретные задания.

В таком исследовании наиболее интересны следующие вопросы:

- стоящая за заданием реальная цель — 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 её 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 заполнения анкеты или проведения интервью, выполняются формальная декомпозиция и анализ рабочих заданий — как правило, при помощи диаграммы потоков или сходной диаграммы, передающей отношения между действиями, часто между людьми и процессами.

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 эффективным инструментом для оценки решений, принятых в ходе проектирования.

Сегментация пользовательской аудитории

Всю массу пользовательских потребностей можно разбить на части путем сегментирования пользовательской аудитории. Деление на различные группы происходит 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 получается 3–4 основные целевые группы.

Для того чтобы лучше понять, какие цели преследуют наши пользователи, сначала мы должны понять, кто они? Чтобы сделать пользователей более конкретными, следует описывать (персонифицировать) типичных представителей каждой группы целевой аудитории.

Персонажи

Персонажи (метод вербализованных представителей) — модели пользователей, создаваемые с целью одушевления процесса разработки. Персонажи — не описания реальных людей, входящих в целевую аудиторию, а некоторые вымышленные представители, наиболее точно описывающие ключевые характеристики целевой аудитории. Персонажи обычно рождаются на стадии проектирования сайта (товара, услуги), а на последующих стадиях живут и развиваются. Придавая облик и имя разрозненным элементам данных, полученным в результате исследования и сегментации пользовательской аудитории, персонажи помогают 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.

Зачастую аудитория потенциальных пользователей представляет собой довольно широкую публику с различными целями и потребностями. Для наиболее чёткого представления основных приоритетов в дальнейшем процессе разработке имеет смысл сегментировать аудиторию потенциальных пользователей по группам. Каждая отдельная группа пользователей преследует свои цели и интересы, которые могут быть как совершенно не похожи на цели других групп, так и переплетаться с ними.

Лекция 1.2. Целевая аудитория, модели пользователей и их цели

Даже самые опытные специалисты-разработчики допускают ошибки. Это правило особенно очевидно в коллективной работе. По мере выполнения проекта накапливаются мелкие ошибки, допущения, неверные решения. В результате получается плохой продукт при хорошей исходной идее. Разработка прототипа — средство, позволяющее проанализировать идеи, прежде чем на них будут потрачены время и деньги. Это распространенный в инженерной практике метод. Основная цель, достигаемая при создании прототипа, — это экономия времени и ресурсов. Ценность прототипа заключается в том, что он является внешней оболочкой-моделью, отражающей существенные отношения разрабатываемого продукта. По сравнению с реальным продуктом прототипы просты и недороги в разработке. При минимальном вложении средств можно обнаружить ошибки создателей и юзабилити проблемы, и улучшить пользовательский интерфейс до того, как сделаны значительные инвестиции в окончательную разработку и технологии.

Создание эффективного прототипа интерфейса является чрезвычайно важной задачей. Прототип должен хорошо выглядеть, чтобы понравиться заказчику и не вызвать вопросов у субъектов тестирования, он должен быть максимально дешёв, максимально полон и, что немаловажно, должен с лёгкостью обновляться.

Требования к прототипу изменяются со временем. Сначала наиболее актуальными его свойствами являются скорость создания и простота модификации. Эти свойства позволяют быстро разработать и проверить несколько версий интерфейса, при этом ещё и исправить значительную часть ошибок.

Затем на первый план выходят функциональность и эстетичность, простота же модификации уже не столь важна, поскольку с каждой новой исправленной ошибкой снижается вероятность того, что прототип придётся полностью переделывать при обнаружении новой ошибки.

Поэтому всегда правильно сделать прототип настолько похожим на результирующую систему, насколько позволяет самая поздняя его версия. Первый прототип стоит делать максимально примитивным. Только после того как тестирование подтверждает его правильность, стоит сделать более детализированный прототип.

Различают четыре версии прототипов: бумажные, презентационные, псевдореальные и реальные. Рассмотрим основные особенности их применения.

Бумажное прототипирование

На первом этапе создания продукта есть только идея и бесконечное множество возможных потенциальных вариантов её реализации. Выбор и материализация их связаны с процедурой гибкого изменения проекта. Вначале, это наброски проекта на бумаге, затем — детальные схемы. Их часто называют бумажными прототипами, хотя и не всегда используется бумага. Скорее это эскизное проектирование с интерактивным исправлением обнаруженных ошибок.

Достоинствами бумаги являются исключительная простота и скорость рисования. Польза начального прототипирования на бумаге заключается, во-первых, в простоте и скорости рисования, во-вторых, в исключительной простоте модификации по результатам тестирования, а в-третьих, в относительной простоте привлечения представителей целевой аудитории. Кроме того, бумага помогает не думать о том, как интерфейс выглядит, позволяя

сосредоточиться на том, как интерфейс работает. Поэтому не следует делать такой прототип очень «реалистичным», он должен просто отражать функциональность. Бумажная версия прототипа рисуется на этапах постановки задачи и высокоуровневого проектирования.

Не следует слишком много использовать бумажную версию для тестирования. Тестирование на бумажном прототипе — проявление неуважения к участнику тестирования. На переключивание бумажек уходит уйма времени; участник при этом скучает, а от скучающего же участника ожидать сотрудничества не приходится. Можно в начале работы провести только пробное тестирование восприятия системы пользователем и её основной логики.

Для создания бумажного прототипа необходимо нарисовать на бумаге все экраны и диалоговые окна (или распечатать соответствующие части схемы). Нужно только убедиться, что все интерфейсные элементы выглядят единообразно и похожи на реальные. Эта распечатка и является первым прототипом для пробного тестирования.

Элементы интерфейса, которые нельзя нарисовать однозначно (например, раскрывающиеся списки, у которых значения до поры скрыты), эффективнее всего рисовать неоднозначными, важную же информацию из них лучше всего словами писать на полях.

Презентационная версия прототипа

После исчерпания возможностей бумажной версии прототипа стоит создать новую (следующую) версию. Для этого точно так же рисуется интерфейс, но уже не на бумаге, а в какой-либо презентационной программе. С этой версией прототипа можно тестировать значительно более сложное взаимодействие человека с системой, нежели с бумажной. С другой стороны, исправление найденных ошибок значительно более трудоёмко. Такая версия прототипа может создаваться уже на этапе высокоуровневого проектирования, а на этапе низкоуровневого проектирования ПИ она обязательна.

Достаточно распространённым инструментом для создания прототипов этого типа является на сегодняшний день MS Visio. При работе с Visio можно выбрать одну из двух возможностей: можно либо рисовать все экраны на одном листе, соединяя взаимосвязанные элементы управления и экраны линиями, либо рисовать каждый экран на отдельном листе, связывая экраны ссылками. Первый вариант удобен, поскольку позволяет оценить интерфейс в целом, второй предпочтительнее для заказчика и субъектов тестирования, поскольку его легче понять. Как правило, превратить второй вариант в первый оказывается гораздо легче.

Большим достоинством Visio является возможность записывать результат в HTML-файл, что позволяет без проблем тестировать интерфейс на территории субъектов тестирования.

В настоящее время практически все программы машинного проектирования имеют возможность создания прототипов. Создано большое количество отдельных пакетов упрощающих жизнь проектировщиков компьютерных интерфейсов и сайтов. Их число постоянно увеличивается, что не позволяет нам выбрать лучшие решения. В качестве примера приведём краткие описания ряда из них.

Mockingbird — очень популярный инструмент для создания прототипов. Он позволяет легко создавать, просматривать и редактировать макет, а так же обмениваться наработками друг с другом. Сервис обладает красивым и удобным интерфейсом, включает

привязку к сетке, перетаскивания, большой набор разнообразных элементов и ряд других полезных функций и возможностей.

Pencil Project — скорее большой плагин для FireFox, нежели онлайн сервис или веб-приложение. После его установки создавать схемы, макеты и пользовательские интерфейсы можно прямо в браузере. Имеется большое количество инструментов и функций (многостраничные документы, поддержка текста, экспорт html, png, файлов OpenOffice и многое другое).

Photoshare предоставляет собой простой инструмент для создания каркасов сайта. С помощью этого сервиса можно создать прототип любой сложности, так как сервис обладает всеми необходимыми инструментами.

HotGloo — очень удобное и популярное средство для разработки макетов. Простой интерфейс содержит широкий набор инструментов, функции и всевозможные средства для создания, редактирования и калибровки прототипа сайта. Различные привязки, функции перетаскивания элементов, обширное количество свойств у объектов. Есть возможность передачи файлов при работе в команде.

MockFlow — онлайн и оффлайн сервис с обширной библиотекой шаблонов и различных элементов, количество которых позволяет свободно управлять разработкой и тем самым делать оригинальные и технически сложные прототипы.

Псевдореальная версия прототипа

В тех случаях, когда в интерфейсе появляются нестандартные элементы или необходимо проверить реальную скорость взаимодействия пользователя с системой, создаётся ещё одна версия прототипа — реально выглядящая, но лишенная каких-либо алгоритмов и, соответственно, не показывающая реальных данных. Делать этот вариант можно как в средах разработки, благо в них есть визуальные инструменты создания интерфейсов, так и в редакторах изображений, что обычно быстрее. Создаются снимки экрана, на которых и производят тестирование. Понятно, что как-то модифицировать эти экраны затруднительно, так что лучше не увлекаться такой работой, не получив каких-либо гарантий её правильности. Эта версия прототипа больше соответствует этапу низкоуровневой разработки ПИ, но может применяться и на этапе высокоуровневой разработки.

Реальная версия прототипа

Иногда необходимо тестировать взаимодействие пользователя не только с интерфейсом системы, но и с обрабатываемыми системой данными. Например, работая с графической программой, пользователь не только нажимает на экранные кнопки, но также создает и модифицирует изображения мышью. Область же редактирования данных зачастую вообще не содержит каких-либо визуальных интерфейсных элементов, из чего вовсе не следует, что интерфейса в ней нет, его наоборот много. Просто счёт в нем идет не на кнопки и переключатели, но на пиксели и миллисекунды.

Понятно, что прототип в таких условиях практически не будет отличаться от готового ПИ. Поэтому лучше всего написать нужные участки программы до написания всего остального и проводить юзабилити-тестирование на реальной версии прототипа ПИ. Понятно, что прототип такой версии, если он все-таки разрабатывается, возможен только на этапе низкоуровневого проектирования.

Лекция 1.3. Сценарии поведения и использования

Что такое юзабилити-тестирование

Юзабилити-тестирование — это набор методов, позволяющих измерить характеристики взаимодействия пользователя с продуктом с целью оценки уровня пользовательских свойств продукта. Как правило, в ходе юзабилити-тестирования изучается, насколько хорошо пользователи выполняют конкретные стандартные задачи и с какими проблемами они при этом сталкиваются. Результаты такого тестирования часто помогают выявить как проблемы, затрудняющие понимание и использование продукта, так и удачные решения.

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

Можно привести определенные доводы в пользу уместности юзабилити-тестирования на ранних стадиях переработки интерфейса существующего продукта. В ряде проектов эта методика позволяет выявить стороны 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 полезнее использовать его как средство проверки конкретных элементов нового дизайна, когда будет готово решение – кандидат версия.

Поскольку результаты пользовательского тестирования обычно измеримы и поддаются количественному выражению, юзабилити исследования продукта особенно ценны при сравнении конкретных вариантов дизайна с целью 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.

Юзабилити-тестирование особенно эффективно при оценке следующих элементов и атрибутов продукта:

- **Наименование**

Осмысленны ли названия разделов и подписи на кнопках? Возможно, какие-то из этих слов воспринимаются легче, чем другие?

- **Архитектура**

Осмысленно ли информация разбита на категории? Расположены ли информационные элементы в тех местах, где их ожидают найти потребители?

- **Первое знакомство и доступность**

Легко ли новые пользователи находят базовые элементы интерфейса? Понятны ли инструкции? Есть ли в них необходимость?

- **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 многократном

использовании продукта. Мешают эффекты обучения пользователя и влияние старого опыта. Один из методов преодоления этих трудностей носит название дневниковые исследования: испытуемые ведут дневники с подробными записями о своём 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 бывает 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 степень детализации, 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 важны для настройки, например, таких аспектов поведения, как скорость прокрутки экрана в качестве реакции на поворот ручки управления. 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.

Есть множество способов проверить интерфейс на реальных пользователях. Их диапазон широк — от неформальных сеансов обратной связи, когда поясняются идеи и выслушиваются соображения пользователя, до более строгого юзабилити-тестирования. 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». В целом подход сеансов обратной связи приемлем для технической аудитории, которая способна домысливать немногочисленные рисунки, представляющие интерфейс продукта. Он может быть достойной альтернативой юзабилити-тесту, если у команды проектировщиков нет времени для формального юзабилити-тестирования.

Если же времени достаточно, необходимо отдать предпочтение более формальному методу [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.

Юзабилити-тестирование по сути своей есть средство анализа, а не синтеза. Оно служит для оценки эффективности существующих идей в качестве инструмента их улучшения, а не для изучения свойств пользователя. 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 исследования, даёт больше возможностей для создания привлекательного продукта. Точно так же, если время и бюджет ограничены, то время, потраченное на проектирование, имеет 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.

Полное и промежуточное тестирование

В своей книге «Usability Engineering» (Nielsen, 1993) Якоб Нильсен различает полное тестирование, когда тестируются завершённые продукты, и промежуточное тестирование, проводимое в ходе проектирования как часть итерационного процесса. Это важное различие.

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. В этот момент уже обычно слишком поздно вносить осмысленные изменения в проектные решения: поезд ушёл. 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, всё же оно относится к сфере информационной поддержки планирования жизненного цикла продукта. Промежуточное тестирование проводится во благо проектирования в ходе процесса проектирования.

Проведение промежуточного юзабилити-тестирования

Существуют различные мнения относительно того, как правильно проводить юзабилити-тестирование и интерпретировать его результаты. К сожалению, мы обнаружили, что многие из подходов либо метят на место проектирования, либо имеют перекося в сторону количественных метрик и дают бесполезные данные вроде «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 по успешному промежуточному юзабилити-тестированию:

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

Вовлеченность проектировщика в юзабилити-исследование

Недопонимание между проектировщиком и пользователем — распространенная причина проблем юзабилити. Персонажи помогают проектировщикам понять цели пользователей, их потребности и их точку зрения, формируя фундамент эффективных коммуникаций. Юзабилити-исследование, открывая дополнительное окно в понимание мышления 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 юзабилитиисследований. 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:

- Планирование исследований — для формулирования важных с точки зрения проектирования вопросов.
- Определение критериев отбора участников с помощью персонажей и их атрибутов.
- Использование сценариев для выбора тестовых задач.
- Наблюдение за сеансами тестирования.
- Совместный анализ результатов исследований.

Следует заметить, что как бы ни совершенны были соображения, приведшие к созданию интерфейса, всегда остается вероятность того, что интерфейс получился плохой, либо, что более вероятно, не таким хорошим, каким бы он мог быть. Необходимо иметь какие-либо подтверждения его работоспособности. Всё, что для этого нужно, это несколько пользователей средней квалификации, плюс прототип (разумеется, при наличии

основательного бюджета можно развернуться, например, купить прибор, фиксирующий направление взора пользователя и длительность зрительных фиксаций).

Юзабилити-тестирование представляет собой постановку экспериментов с целью выявления информации, касающейся эргономики и дизайна ПИ. Самый адекватный перевод слова «usability» — это «потребительские качества продукта». Существо идеологии юзабилити-тестирования — придать ПИ максимально высокие потребительские свойства (удобство, продуктивность, легкость понимания, легкость обучения). Короче говоря — «все во имя пользователя и на благо пользователя». Юзабилити-тестирование — стержень эргономической экспертизы ПИ.

Существует распространенное мнение, что с помощью тестирования можно решить все проблемы интерфейса. Это не верно. 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 разрабатываемого продукта.

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.

Подготовка к тестированию

Прежде всего, надо очень ясно поставить вопрос тестирования, причём 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 тестирования, его, так сказать, композицию, которая описывает, как вы будете проводить отдельные тесты, в каком порядке они будут выстроены, чтобы исключить из рассмотрения и дальнейшего анализа все переменные, не представляющие интереса.

Разрабатываются задания, которые будут предложены участникам тестирования. Эти задания, несомненно, 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 вам нужен и приобрести его. Инструментарий может включать в себя устройства, используемые в процессе проведения тестов:

- видеокамеры для записи поведения пользователей;

- преобразователи развертки для 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 так далее.

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 выборку в требуемой для теста пропорции характеристик пользователей (скажем, опыта, навыков и демографических характеристик), 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. Тем не менее, для тех целей, которые преследуете вы («Позволяет ли использование дисплея радар ближайшего окружения самолёта избежать столкновений в процессе полета на спине?»), вы можете быть заинтересованы в рассмотрении только одного сегмента всей выборки — в нашем примере — только пилотов.

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 давлением. Обязательно следует поблагодарить пользователей за участие в тестах. Надо объяснить участникам, что они могут в любой момент остановить тест, отлучиться в туалет 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 чём они думают, можно спросить у них об этом.

Необходимо понять, какие ожидания возникают у пользователей на каждом шагу и насколько программа соответствует 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 щёлкнуть мышью, можно спросить у него, что он ожидает увидеть. После того как он щёлкнет 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 (чего, 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 вы раньше думали, но по каким-то причинам отклонили. 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.

Анализ полученных данных

После того как задания выполнены, и тестирование завершено, следует обсудить тест с его участником. Это необходимо для того, чтобы получить дополнительную информацию, касающуюся того, о чём 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 должен как можно быстрее написать небольшой отчёт об основных проблемах, которые были замечены во время тестирования, а также изложить свои мысли по поводу способов их устранения. Не надо писать полных и развернутых отчётов. Пусть это напоминает скорее резюме. В идеале, все участники группы по разработке должны прочитать эти отчёты (или хотя бы пробежать их глазами), поэтому они не должны по объёму превышать 1–2 страницы.

На последующем собрании рабочей группы следует обсудить два вопроса:

- Каковы проблемы, с которыми сталкивались пользователи, и какие из них должны быть исправлены?
- Каковы возможные решения отобранных проблем?

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

Данным, касающимся производительности, таким, как частота ошибок и времена выполнения заданий, оценка даётся с помощью статистического анализа. Большая часть такого анализа сводится к нахождению среднего значения и стандартного (среднего квадратичного) отклонения, а также к проверке достоверности полученных данных.

Наблюдения за действиями пользователей и запись их мнений как во время теста (используя метод записи «мыслей вслух» или задавая вопросы), так и до или после теста проводятся с использованием анкет и опросных листов. Большинство таких опросников имеют строение, позволяющее количественно измерить мнения, используя численную шкалу, и полученные количественные данные могут анализироваться также с помощью статистических методов.

Модуль 2. «Элементы пользовательского интерфейса»

Лекция 2.1. Структура интерфейса

Языки, уровни и виртуальные машины

С помощью органов управления (пульта) автоматизированного рабочего места (АРМ) оператор осуществляет исполнительные и управляющие действия необходимые для штатного функционирования обслуживаемой им системы. Грамотное проектирование органов управления (ОУ) должно обеспечивать оператору возможность быстрого нахождения необходимого исполнительного органа управления и выполнения требуемого действия с заданной точностью в пределах допустимого интервала времени.

Поиск способов эффективного управления АРМ требует учёта большого количества вариантов разнообразных алгоритмов деятельности операторов, переменных параметров среды отображаемых в виде цифр, букв, графики. Обеспечение надёжной работы АРМ требует концентрации на управлении режимами системы, которые осуществляются в АРМ с помощью различных органов управления:

- экранного — управление кнопками и пиктограммами, отображёнными на экране;
- физического — управление с помощью физических органов управления (кнопок, тумблёров, переключателей);
- с помощью графического манипулятора (трекбол, мышь и др.).

Количественное соотношение физических, экранных переключателей и управляющих воздействий графическим манипулятором для каждого вида АРМ определяются экспертным путём и зависят от специфики реализуемых алгоритмов управления и традиций проектирования, принятых в конкретной разрабатывающей СЧМ организации.

Основой для выбора данного соотношения служат требования системы по количеству управляющих воздействий, их скорости и точности выполнения. При этом учитывается, что сенсорная нагрузка на оператора распределяется неравномерно. Большую часть информации оператор получает через зрительный анализатор. Далее следуют слуховой и тактильный анализаторы.

Стремление замены физических ОУ на экранные ОУ приводит к большей загрузке зрительного анализатора и к бездействию левой руки (экранное управление осуществляется с помощью графического манипулятора обычно правой рукой). Возникает проблема перераспределения потоков информации передаваемой оператору между различными анализаторами с целью снятия перегрузки со зрительного анализатора. Она решается использованием полимодального представления информации через анализаторные системы человека: слуховую и тактильную.

Слуховой анализатор является филогенетически одним из наиболее рано сформировавшимся, а потому и наиболее устойчивым к внешним воздействиям. Он адекватно отражает рабочую среду, когда работа зрительного анализатора затруднена: в условиях кислородного голодания на больших высотах, при воздействии больших положительных ускорений и т. п. Он обладает большим диапазоном частот (от 16–20 до 20–22000 Гц.) и интенсивностей (до 130–140 дБ.), независим от пространственного положения, обладает высокой помехоустойчивостью. В то же время к его недостаткам относится то, что он, являясь анализатором времени, воспринимает информацию последовательно, а не симультанно, как зрение, и поэтому обладает невысокой скоростью и пропускной способностью. В силу этого оперативная память оператора оказывается перегруженной.

Слуховую форму предъявления информации следует использовать в следующих случаях:

- для сигналов опасности, так как слух, в отличие от зрения, не способен к произвольному самовыключению;
- при перегрузке зрения;
- когда работа оператора требует его постоянного перемещения и информация должна приниматься независимо от ориентации головы оператора;
- при ограничении зрения внешними или внутренними условиями;
- в специфических условиях (аноксия, состояние невесомости, воздействие перегрузок и т.п.);
- когда в сообщениях системы речь идёт о событиях, разворачивающихся во времени;
- при необходимости выделения сигнала из шума, так как слуховой анализатор — хороший детектор периодических сигналов на фоне шума.

Различают звуковые и шумовые сигналы, с одной стороны, и речевые — с другой. Использование звуковых и шумовых сигналов рекомендуется в следующих случаях:

- при приёме простого и короткого сообщения, не связанного с последующими сообщениями;
- когда сообщение требует немедленного действия;
- когда оператор специально обучен пониманию смысла закодированного сообщения;
- если оператор перегружен речевыми сигналами;
- если необходимо соблюдение тайны;
- когда оператор работает в группе;
- при сильных акустических помехах.

Звуковое предъявление информации используется во всех гидролокационных системах для определения и обнаружения контуров объектов по отражённому звуку.

Тактильный анализатор воспринимает ощущения, возникающие при действии на поверхность кожи различных механических стимулов — прикосновение, давление, вибрация

Абсолютный порог каждой чувствительности определяется по минимальному давлению предмета на кожную поверхность, которая производит едва заметные ощущения прикосновения. Примерные пороги ощущения: для кончиков пальцев руки — 3 г/мм^2 ; на тыльной стороне пальца — 5 г/мм^2 ; на тыльной стороне кисти — 12 г/мм^2 ; на животе — 26 г/мм^2 ; на пятке — 250 г/мм^2 . Временной порог тактильной чувствительности — менее 0,1 с. Тактильный анализатор позволяет на ощупь определять характерные формы органов управления (рукояток, кнопок, тумблеров) и облегчать или ускорять процесс управления в ухудшенных условиях зрения. Кроме того, может использоваться дополнительный вибрационный канал информации. В настоящее время вибрация используется в сенсорных клавиатурах и при управлении мобильными устройствами (сотовыми телефонами, коммуникаторами, планшетными компьютерами). Перспективно использование тактильных каналов обратной связи для усиления чувства присутствия в виртуальной реальности (перчатки и жилеты виртуальной реальности).

При выборе видов органов управления учитываются формы и алгоритмы деятельности оператора и требования системы.

Разнообразие задач, решаемых операторами и жёсткие требования по надёжности их выполнения, требуют разработки специализированных физических и экранных ОУ. Самые популярные из них — клавиатуры.

Спецификация клавиатур включает: количество кнопочных переключателей, их номенклатуру и форму исполнения (физическую или экранную), их компоновку, наименование. Класс клавиатуры, её форма и цветовое решение должны быть

унифицированы для всех АРМ, имеющихся в системе. При проектировании специализированных клавиатур учитываются общие эргономические требования к органам управления.

Эргономические требования к физическим органам управления

Устанавливаются государственными стандартами по эргономике. Например, для промышленного оборудования действует ГОСТ 12.2.049-80 — Система стандартов безопасности труда. Оборудование производственное. Общие эргономические требования (Occupational safety standards system. Industrial equipment. General ergonomic requirements).

В соответствии с ним конструкция органов управления должна учитывать:

- требуемую точность и скорость движений при осуществлении управления, а также частоту использования органа управления;
- допустимые динамические и статические нагрузки на двигательный аппарат человека;
- антропометрические характеристики двигательного аппарата человека;
- необходимость быстрого распознавания органов управления, формирования и закрепления навыков по управлению.

При конструировании органов управления и их размещении в моторном поле рабочего места должны быть учтены следующие физиологические особенности двигательного аппарата человека:

- скорость движения рук больше при движении в направлении «к себе», меньше — при движении «от себя»;
- скорость движения правой руки больше при движении слева — направо, левой руки — справа — налево;
- линейная скорость вращательных движений рук больше скорости поступательных движений;
- скорость плавных криволинейных движений рук больше скорости прямолинейных движений рук с резким изменением направления;
- точность движения рук больше при работе в положении сидя, меньше — при работе в положении стоя.

Усилия, необходимые для осуществления управляющих действий, должны устанавливаться с учётом способа перемещения органа управления (пальцами, кистью с предплечьем, всей рукой, стопой и т.д.), частоты использования и в некоторых случаях с учётом продолжительности непрерывного воздействия на органы управления, скорости выполнения управляющего действия и положения человека в процессе управления.

Для обозначения функционального назначения органов управления следует применять надписи и (или) символы, которые должны быть расположены на элементах конструкции рабочего места в непосредственной близости от органов управления или на их приводных элементах. Органы управления должны кодироваться формой, цветом, размером или другими видами алфавита кода или их комбинациями.

Наиболее массовым органом управления в компьютерах является кнопочная клавиатура. В условиях интенсивной нагрузки на зрительный анализатор она обеспечивает не только зрительное, но и слуховое и тактильное опознание момента нажатия на клавишу.

Основные эргономические требования к клавиатуре.

Конструкция клавиатуры должна предусматривать:

- исполнение в виде отдельного устройства с возможностью свободного перемещения;
- опорное приспособление, позволяющее изменять угол наклона поверхности клавиатуры в пределах от 5° до 15°;

- высоты среднего ряда клавиш не более 30 мм;
- расположение часто используемых клавиш в центре, внизу и справа, редко используемых — вверх и влево;
- выделение цветом, размером формой и местом расположения функциональных групп клавиш;
- минимальный размер клавиш — 13 мм, оптимальный — 15 мм;
- клавиши с углублением в центре и шагом 19 ± 1 мм;
- расстояние между клавишами не менее 3 мм;
- одинаковый ход для всех клавиш с минимальным сопротивлением нажатию 0,25 Н и максимальной — не более 1,5 Н;
- звуковую обратную связь от включения клавиш с регулировкой уровня звукового сигнала и возможностью его отключения.

Для кодирования информации в кнопочных переключателях оптимальным является использование от 2-х до 6-ти различительных признаков: форма, цвет, размер, местоположение кнопки, ощущение слышимого щелчка (клика), вертикальное перемещение.

В настоящее время в массовых масштабах выпускаются кнопочные физические клавиатуры на трёх разных технологиях: клавишные, плёночные (мембранные) и резиновые.

Каждая технология имеет свои достоинства и недостатки.

Клавишные клавиатуры удовлетворяют эргономическим и эксплуатационным требованиям, но требуют конструкторской доработки мест размещения при перекомпоновке переключателей при создании специализированной клавиатуры.

Плёночные клавиатуры удовлетворяют всем эргономическим и эксплуатационным требованиям. Их достоинства:

- тонкие, легко чистящиеся;
- пыле-, влаго- и маслозащищённые;
- герметичные;
- обладают привлекательным внешним видом;
- возможно различное конструктивное и цветовое исполнение;
- имеют низкую цену.

Резиновые клавиатуры часто не удовлетворяют условиям эксплуатации, особенно плохо работают в химически-агрессивных средах и при значительных температурных колебаниях.

При размещении физических ОУ на панели управления следует учитывать физиологические особенности двигательного аппарата человека, определяющие усилие, точность и скорость движения его рук. ОУ следует размещать в зонах досягаемости моторного поля, определяемых значениями антропометрических характеристик предполагаемого контингента операторов.

Эргономические требования к графическим манипуляторам

Основным средством управления в эргатических системах, включающих компьютеры, является графический манипулятор (ГМ), с помощью которого происходит управление экранными органами управления. Манипуляторы осуществляют непосредственный ввод информации, указывая курсором на экране монитора команду или место ввода данных. Манипуляторы, как правило, подключаются к последовательному порту. Используются для облегчения управления компьютером (ПК). К манипуляторам относятся мышь, клавиатура, трекбол, графический планшет (дигитайзер), световое перо, тачпад, сенсорный экран, Roller Mouse, pointing stick, джойстик и игровые манипуляторы.

Манипуляторы классифицируют на следующие типы:

- с относительным указанием позиции (перемещения): мышь, трекбол, трекпоинт, тачпад, джойстик, Roller Mouse;

- с возможностью указания абсолютной позиции: графический планшет, световое перо, аналоговый джойстик, клавиатура;

- игровые манипуляторы: джойстики, геймпады, компьютерный руль, танцевальная платформа, сенсор Kinect.

О последнем устройстве нужно сказать отдельно. Оно открывает новый класс устройств — игровой «контроллер без контроллера». Kinect позволяет пользователю взаимодействовать со средой игровой приставки без помощи игрового контроллера через устные команды, позы тела и показываемые объекты или рисунки. Это небольшое устройство, в которое вмонтированы две камеры высокого разрешения, сканирующие трёхмерное пространство с датчиками распознавания голоса. Камеры позволяют устройству видеть глубину, т.е. устроены наподобие человеческого стереоскопического зрения, с помощью которого мы и оцениваем объёмы, расстояния, размеры. Также, эти камеры достаточно точны, чтобы отличить одного человека от другого. Манипулятором является само тело играющего.

Основным требованием для графического [манипулятора](#) является быстрое и точное перемещение указателя манипулятора по экрану в заданную точку (точность наведения должна составлять 0,3 мм, и достигаться с одной попытки, т. е. итеративное приближение к заданной на экране точке — нежелательно).

В настоящее время в промышленном проектировании наиболее популярны три класса графических манипуляторов, которые можно использовать для создания систем управления в эрготехнических комплексах: сенсорный, трекбол и джойстик.

Примерами технических реализаций манипуляторов из данных классов могут служить:

1. Сенсорный ГМ — 1, разработан КБ «КОРУНД-М» (Москва). Тип «БАГЕТ» — изготовитель г. Пенза. Устройство чувствительно к прикосновению пальца. Имеет 4 кнопки для правой и левой руки оператора. Удовлетворяет условиям эксплуатации.

2. Трекбол

- 2.1. Манипулятор шаровой (Трекбол). Разработан в КБ «КОРУНД -М». Тип БАГЕТ — изготовитель г. Пенза. Имеет три кнопки для правой и левой руки оператора. Удовлетворяет условиям эксплуатации.

- 2.2. Трекболы типа ТВБ34, TBL50, TBS50F. Изготовитель фирма NSI, Бельгия. Шар диаметром соответственно 34 и 50 мм. Разрешение 200 точек на оборот.

- 2.3. «Трекбол» — ЦНИИ «Агат» (Москва). Диаметр шара 50 мм. Удовлетворяет условиям эксплуатации.

3. Джойстик. Изготовитель NSI, Бельгия. Тип управления — перемещение рычага и тактильные кнопки.

Графический [манипулятор](#) должен быть унифицирован для управления в АРМ на всех системах организации. Сенсорные графические манипуляторы, использующие палец, если не предусматриваются специальные формы контактного устройства, требуемой точности наведения обеспечить в обычных условиях не могут. Точность наведения может быть достигнута только при специальной тренировке и профотборе. Так, например, наиболее приемлемыми для задач, решаемых на АРМ систем военно-морского флота, являются трекболы.

Эргономические требования к экранным органам управления

Основными экранными ОУ являются: экранная кнопка, контекстное меню, поле со списком, линейка прокрутки, текстовое окно, указатель визир-маркер.

- экранная кнопка — экранный переключатель с 2 положениями вкл/выкл.
- контекстное меню — экранный переключатель типа «оконного меню».

- поле со списком — таблица данных, которые можно выбрать с помощью графического манипулятора и изменить с помощью физической клавиатуры.
- линейка прокрутки — шкала на экране с маркером, управляемым графическим манипулятором.
- текстовое окно — поле для набора информации с помощью физической клавиатуры,
- указатель визир-маркер — различного вида точечные \ линейные указатели, управляемые графическим манипулятором.

Унификации экранных органов управления подлежат формы органов управления, их представление, цветовая палитра, шрифты.

Все остальные параметры экранных органов управления унифицируются в рамках АРМ одной системы. При этом учитываются общие эргономические требования к экранным ОУ.

Лекция 2.2. Принципы визуального дизайна и идиомы в пользовательских интерфейсах

Основные функции и требования

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 уменьшают множество возможных вариантов построения интерфейса.

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

1. Управления компьютером путём действий оператора (пользователя): инициация, прерывание, отмена компьютерных процессов и т. п.
2. Ввод данных, осуществляемый оператором, и отклик системы.
3. Отображение данных, включающее отображение данных, вводимых оператором, который может управлять процессом отображения данных.
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 всестороннее использование потенциальных возможностей оператора, технических и информационно-программных средств АРМ, высокие безошибочность и быстродействие деятельности оператора при применении ПИ по назначению. Хорошо спроектированный ПИ обеспечивает максимальный комфорт деятельности оператора и должен:

- способствовать быстрому освоению вычислительной техники оператором, формированию у него устойчивых навыков;
- быть спроектирован таким образом, чтобы оператор вводил информацию естественным образом, не заботясь о ходе вычислительного процесса;
- удовлетворять рабочие потребности человека-оператора, а не обслуживать процесс обработки данных. Синтаксическая структура, реализованная в интерфейсе должна:
 - соответствовать ожиданиям оператора решающего профессиональную задачу;
 - содержать систему правил работы оператора, обеспечивающую лёгкое управление системой;
- постоянно находиться под контролем оператора, никакие действия последнего не должны приводить к тупиковой ситуации или зависанию программы. Интерфейс и справочный механизм информационной системы должны:
 - обеспечивать возможность лёгкого исправления ошибок ввода, не должен требовать повторного ввода данных;
 - обеспечивать оператора информацией, позволяющей ему управлять диалогом, распознавать и исправлять ошибки, определять последующие действия, входящие в алгоритм;
 - обеспечить конкретность и понятность оператору выдаваемой компьютером информации;
 - согласование объёма представляемой оператору информации с возможностями его кратковременной памяти;
 - в информации об ошибках оператора делать акцент не на неправильные действия последнего, а на то, чем и как можно исправить ошибки;

- предусматривать использование четырех видов диалога: меню, команды, манипуляции и диалог посредством заполнения форм.
- при решении каждой задачи предоставить оператору возможность использования не менее двух видов диалога. Критерием выбора вида диалога в ходе решения конкретной задачи является обеспечение заданных показателей безошибочности и быстродействия.

Структура пользовательского интерфейса

Пользовательский интерфейс состоит из трёх основных частей:

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 зачастую разработчики программных продуктов рассматривают функциональность системы отдельно от её пользовательского интерфейса, и редко 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 её внедрения, а также полностью использовать заложенную в ПО функциональность.

Стили пользовательского интерфейса

Существует ряд стилей пользовательского интерфейса, которые завоевали популярность в индустрии программных средств. Основные виды стилей ПИ представлены на Рис. 1

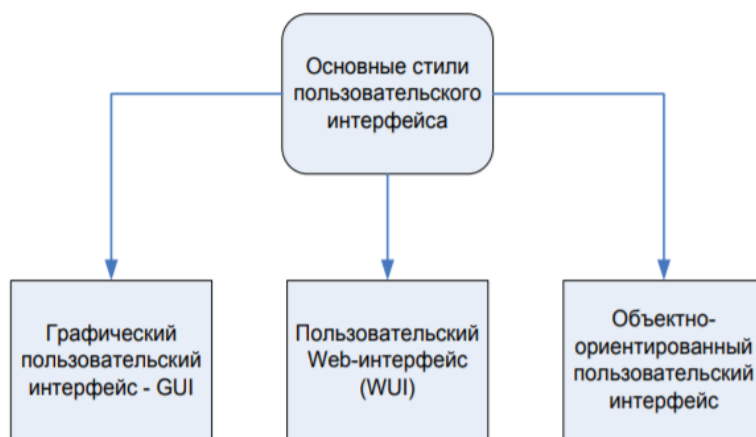


Рис.1. Основные стили пользовательского интерфейса

Самыми популярными являются GUI-интерфейсы (GUI – Graphical User Interface) и построенные на их основе WUI-интерфейсы (WUI – Web User Interface). Стилиевые детали WUI-интерфейсов незначительно отличаются от GUI-интерфейсов, подтверждением чему служат диалоговые окна Web-браузеров.

Графический ПИ (GUI-интерфейс)

Графический пользовательский интерфейс (Graphical User Interface – GUI) определяется как стиль взаимодействия «пользователь – компьютер», в котором применяются четыре базовых элемента: окна, пиктограммы, меню и указатели (рис. 2). Иногда GUI-интерфейс называют WIMP-интерфейсом (Windows – окна, Icons – пиктограммы, Menus – меню и Pointers – указатели).

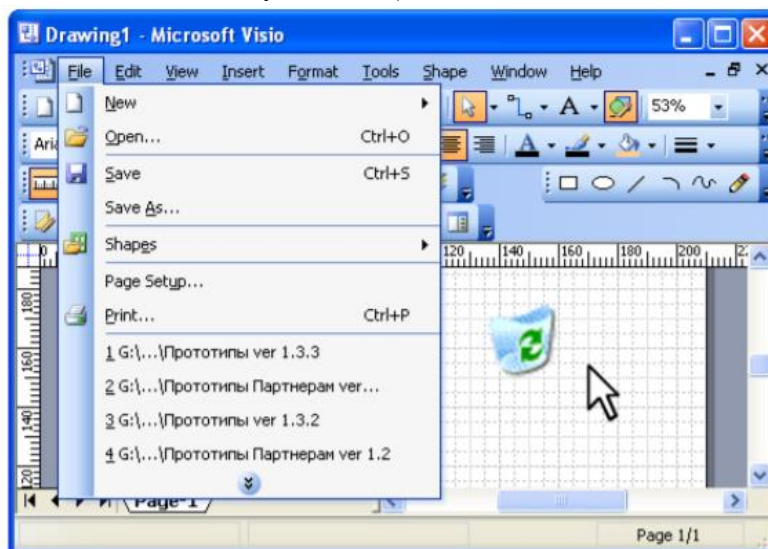


Рис. 2. Пример GUI-интерфейса

Важнейшие свойства GUI-интерфейса — это возможность непосредственного манипулирования, поддержка мыши или указателя, использование графики и наличие области для функций и данных приложения. Начальный анализ основ GUI-стиля ведётся отдельно от прикладного уровня GUI-ориентированных приложений. Непосредственное манипулирование. Наиболее значительное свойство GUI-интерфейса заключается в непосредственном манипулировании, которое позволяет пользователю взаимодействовать с объектами с помощью указателя. Например, окно можно переместить по экрану с помощью мыши, установив указатель на строку заголовка окна, нажав и удерживая кнопку мыши и перемещая мышь (иногда эту операцию называют «захватить и перетащить» — «grab and drag»). Другой пример непосредственного манипулирования с помощью указателя — это выделение текста («занять [место] и ввести» — «swipe and type») или рисование непосредственно в графической области с использованием указателя и графических инструментов наподобие кисти (paint brush).

Многие действия, выполняемые с помощью выбора альтернатив или меню, можно произвести, воспользовавшись непосредственным манипулированием. Например, во многих системах результатом перетаскивания пиктограммы документа на пиктограмму принтера на рабочем столе является печать документа. К другим действиям, которые выполняются с помощью непосредственной манипуляции, относятся такие операции, как Move (Переместить), Copy (Копировать), Delete (Удалить) и Link (Связать).

Другие свойства. К некоторым другим методам работы ПИ, присущим GUI-интерфейсу, относятся буфер обмена, комбинации клавиш, ускоряющие клавиши в меню и диалогах, а также дополнительные возможности взаимодействия мышь-клавиатура.

Несмотря на свою полезность, эти механизмы не рассматриваются как существенные свойства GUI-интерфейса.

Пользовательский Web-интерфейс (WUI-интерфейс)

Базовый WUI-стиль (Web User Interface) весьма схож с меню иерархической структуры, которое пользователи знают по опыту работы в средах с графическим интерфейсом за исключением более наглядного представления и использования гиперссылок. Необходимая навигация выполняется в рамках одного или нескольких приложений с использованием текстовых или визуальных гиперссылок. В зависимости от структуры гиперссылок приложения навигация в пределах WUI-интерфейса приводит к отображению Web-страниц в иерархии приложения — по одной за раз — в линейном или нелинейном стиле внутри одного [GUI-окна](#). Во многих отношениях WUI-ориентированные приложения — это «шаг назад в будущее» — или, может быть, нечто худшее, учитывая объёмы электронных документов и других материалов в формате Web.

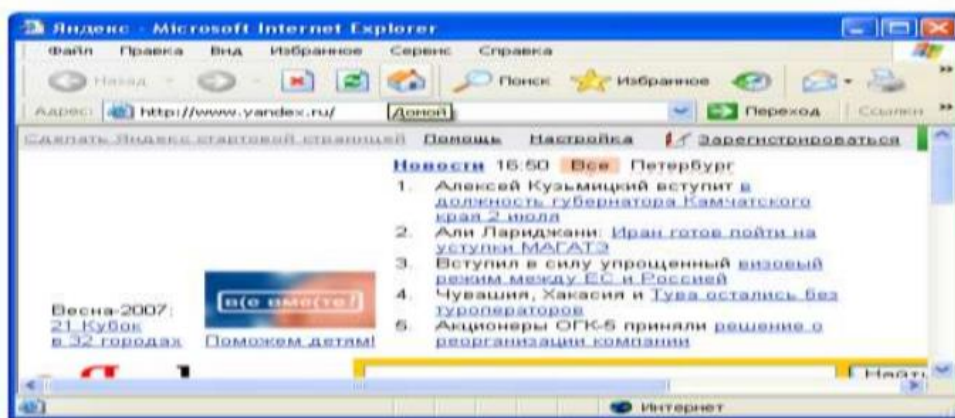


Рис. 3. Пример WUI-интерфейса — Web-браузер Internet Explorer

Основные особенности приложения, использующего WUI-стиль:

- Информация обычно отображается в единственном GUI-окне, называемом Web-браузером, хотя для представления данных в приложении могут использоваться несколько окон.
- Web-браузер обеспечивает меню для Web-приложения.
 - о Выбор действий ограничен, так как меню, обеспечивающее обращение к функциям, не является легкодоступным для приложения.
 - о Web-страница обладают небольшой степенью внутреннего контроля над клиентской областью для открытия специализированных всплывающих меню.
 - о Создание специализированных меню требует дополнительной работы по программированию.
 - о Функциональные возможности приложения должны отображаться в методы для вызова команд.
- Клиентская область не содержит традиционные пиктограммы.
 - о Многие приложения используют графику и анимацию в эстетических или навигационных целях. Это таит в себе потенциальную угрозу возникновения внешнего визуального шума и увеличения времен отклика при загрузке и раскрытии графических файлов.

о Web-браузер и приложения обеспечивают возможности отключения графики, содержащейся в Web-страницах, так что на экране отображается только их текстовая версия.

- Поддержка указателя осуществляется в основном для выбора с помощью одного щелчка мышью или выбора по навигационным ссылкам. Технология «drag and drop» («перетащить и поместить») не поддерживается за исключением случаев специального программирования в определенных средах.

Web-ориентированное ПО становится все более похожим на GUI-ориентированное ПО (возможно потому, что пользователи неизменно требуют наличия популярных и полезных свойств GUI-интерфейса наподобие функции «drag and drop» или всплывающих меню).

Навигация. Переход от одной страницы к другой с использованием гиперссылок или поискового механизма — наиболее часто выполняемая функция WUI-интерфейса. Страницы, с которыми встречается пользователь, существуют в пределах того же самого или другого Web-узла.

Web-браузер обеспечивает базовые возможности навигации для перемещения по Web-узлам и в пределах Web-узлов линейным способом с помощью кнопок панели инструментов Back (Назад) и Forward (Вперед). Навигация от одной страницы приложения к другой в пределах одного и того же Web-узла приложения выполняется с использованием гиперссылок, схемы Web-узла, кнопок и навигационной панели.

Представление и поведение. Основное назначение Web-страницы заключается в обеспечении полезной информацией, включая навигационную структуру и организацию Web-узла. Web-страницы составлены из одной или нескольких конструкций, представляющих собой сочетание мозаик цветных графических элементов. По сравнению с GUI-ориентированными приложениями WUI-ориентированные приложения включают большое количество элементов поведения, которые не вызываются пользователем, например, анимационных. С точки зрения ПИ в Internet царит полная анархия.

Компоненты WUI-интерфейса. К наиболее распространенным компонентам WUI-интерфейса относятся баннеры (заголовки), навигационные панели и визуальные или текстовые гиперссылки, упорядоченные различными способами. Также применяются разнообразные подходы к использованию графики, анимации и цвета.

- Баннер представляет собой визуальный заголовок, отображаемый сверху Web-страницы.

- Навигационная панель — это список вариантов выбора гиперссылок, обеспечивающих доступ к информации сайта.

- Гиперссылка. Представляет собой вариант выбора, который отображает следующую страницу информации или перемещает фокус отображения на другую область той же страницы. Макеты Web-страниц. Информация представляется на Web-страницах с использованием одного или нескольких макетов и навигационных стилей.

- Браузер. Типичный браузер обладает заголовком, навигационной панелью и областью, отображаемой в пределах экрана.

- Каталог. Каталог представляет собой визуальный поисковый механизм, в котором перечислены варианты выбора гиперссылок, используемых для навигации по дополнительным вариантам выбора до тех пор, пока не будет найден искомый результат. Допускаются навигационные панели в виде заголовков и другие типы навигации по вариантам выбора гиперссылок.

- Поиск и результаты поиска. Один или несколько элементов управления, с помощью которых пользователь осуществляет ввод или выбор критерия поиска информации. Результаты поиска отображаются в том же или другом окне Web-браузера.

- Документ. Во многом похожий на свой бумажный двойник Web-документ отображает текстовую информацию вместе со ссылками на дополнительные источники или развернутое представление информации.

- Записная книжка. Некоторые Web-узлы представляют визуальную записную книжку в качестве метафоры для организации данных. Она почти не отличается от навигационной панели, с той лишь разницей, что содержит меньшее количество вариантов выбора.

Сбор данных. Дополнительной целью некоторых Web-страниц является сбор имен и адресов, а также другой информации о пользователях. для этого используются обычные элементы управления GUI-интерфейса, иногда с некоторыми ограничениями.

Проблемы проектирования. Факторы успешного проектирования, которые влияют на качество приложений, использующих Web-стиль ПИ: простота навигации по иерархическим информационным структурам, лёгкость и быстрота поиска, а также быстрая реакция. К другим важным факторам относятся эстетические характеристики и ценность текущего содержания информации.

Объектно-ориентированный пользовательский интерфейс

Проектирование программных объектов даёт возможность предоставить в распоряжение пользователя приложение, обладающее объектно-ориентированным стилем ПИ и/или объектно-ориентированной внутренней структурой (реализацией). Многие объектно-ориентированные свойства реального мира находят отражение во внешнем виде, поведении, требованиях к взаимодействию и функциональных возможностях. Компьютеризованное усовершенствование или дополнение объектов реального мира, если только оно плохо спроектировано или реализовано, не очевидно для конечного пользователя и не в состоянии преодолеть его устоявшиеся знания и восприятие. Представленные в явном виде при проектировании, обозначения классов объектов, иерархии классов и наследование посредством иерархии классов остаются прозрачными для пользователя.

Объектно-ориентированный прикладной ПИ должен обладать следующими свойствами:

- обеспечивать непосредственное манипулирование (перетаскивать любые объекты куда угодно);
- обеспечивать непосредственный ввод данных (записывать любую информацию);
- обеспечивать контекстную зависимость от объектов (всплывающие (контекстные) меню, справки, согласованность и т. д.).

Хороший прикладной объектно-ориентированный ПИ прост в использовании; это значит, что его механизмы пользовательского интерфейса прозрачны.

Лекция 2.3. Проектирование и доработка на примере стандартных элементов интерфейса

Для получения эффективного результата разработки ПИ интерфейса используют различные подходы к проектированию:

1. Подход, ориентированный на пользователя (User Centered) — основным содержанием этого подхода является ориентация на пользователя, т. е. в первую очередь необходимо узнать, что хочет пользователь получить от проектируемого интерфейса. Далее в процессе проектирования полученные требования реализуются в продукте. При сборе информации используются методы наблюдения за работой пользователя, проводятся интервью.

2. Системный подход (System). Пользователь рассматривается как маленькая интеллектуальная часть системы «человек – программный продукт».

3. Деятельностный подход (Activity Centered). Изучается деятельность пользователя в целом, и постепенно оптимизируются её отдельные моменты.

4. Итеративный подход (Agile) — метод последовательных приближений. Суть итеративного подхода заключается в создании изначально самого простейшего прототипа с целью показать заказчику и затем постепенно дорабатывать прототип, основываясь на реакции заказчика после каждого шага доработки.

5. Экспертный подход (Genius). Заключается в следующем: эксперт собирает важную, по его мнению, информацию, ведёт переговоры с заказчиком, задаёт нужные вопросы. На основе полученной информации создаётся интерфейс.

6. Целеориентированный подход проектирования (Goal Centered Design). Разработка интерфейса ориентируется на цель, которая будет достигаться данным программным продуктом.

7. Средаориентированный подход. Разрабатывается среда интерфейса как место деятельности оператора. При разработке интерфейса целесообразно гибко пользоваться указанными подходами, учитывая при выборе методов: назначение разрабатываемого продукта, целевую аудиторию, время и бюджет разработки.

Этапы эргономического проектирования интерфейса

Проектирование интерфейса представляет собой довольно сложный и многоэтапный процесс, каждый этап которого состоит в свою очередь из отдельных ступеней. В общем случае весь процесс можно представить в виде четырёх этапов (рис. 4), направленных на решение основной задачи — обеспечить оптимальное взаимодействие пользователя с системой.

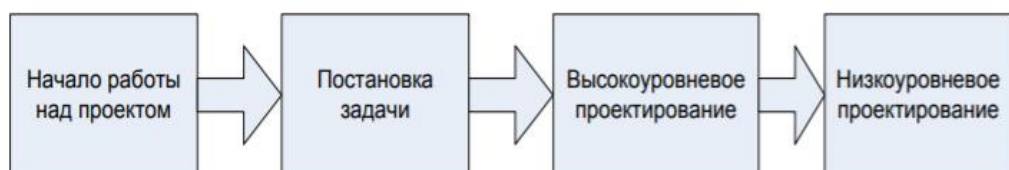


Рис. 4. Этапы эргономического проектирования пользовательского интерфейса

При решении данной задачи нельзя рассматривать составляющие её части отдельно друг от друга. В системе «человек – машина» главным звеном является человек, а система, являясь подчиненным звеном, должна реагировать на его действия, а не наоборот. В связи с данным положением, возникает задача проектирования опыта взаимодействия.

Проектирование опыта взаимодействия или проектирование взаимодействия — это новая область научно-практической деятельности, которая в последние годы выделяется как

самостоятельная дисциплина, сосредоточенная на проектировании поведения пользователя продукта.

Проектирование взаимодействия — это описание возможного поведения пользователя и определение того, как система будет реагировать на его поведение, и приспосабливаться к нему.

Потребовалось немало времени, прежде чем разработчики пришли к мысли, что нужно перейти от разработки программного обеспечения, хорошо работающего с точки зрения машины, к созданию программ, хорошо работающих с точки зрения человека.

Проектирование взаимодействия касается не столько эстетических аспектов, сколько понимания потребностей пользователей и принципов их познавательной деятельности. Форма и эстетическая привлекательность продукта должны работать в гармоничной связке при достижении целей пользователей посредством правильно спроектированного поведения продукта.

Именно опыт взаимодействия с программным продуктом формирует у пользователя впечатление о товарах или услугах, предлагаемых компанией, именно опыт взаимодействия отличает компанию от её конкурентов, определяет — будет ли пользователь в дальнейшем работать с нашим продуктом или предпочтёт конкурента.

Положительный опыт взаимодействия является результатом множества крупных и мелких проектных решений о том, как продукт выглядит, как себя ведёт, и какие действия позволяет совершать пользователям. Эти решения опирающиеся друг на друга, формируют опыт взаимодействия и влияют на все его аспекты.

С переходом на каждый последующий [уровень](#) вопросы, на которые разработчики ищут ответы, становятся всё более конкретными. На нижнем уровне мало кто думает об окончательном внешнем виде продукта. Волнует лишь то, насколько разрабатываемый продукт впишется в стратегию проектирования, при этом удовлетворяя потребности пользователей. От этапа к этапу (уровня к уровню) решения становятся более конкретными и обретают новую степень детализации.

Каждый уровень зависит от уровней, расположенных ниже: поверхность зависит от компоновки, которая зависит от структуры, которая зависит от набора возможностей, которые в свою очередь зависят от стратегии.

Данный подход представлен в виде этапов (рис. 5). Обобщая всё вышесказанное, рассмотрим каждый из них более детально.

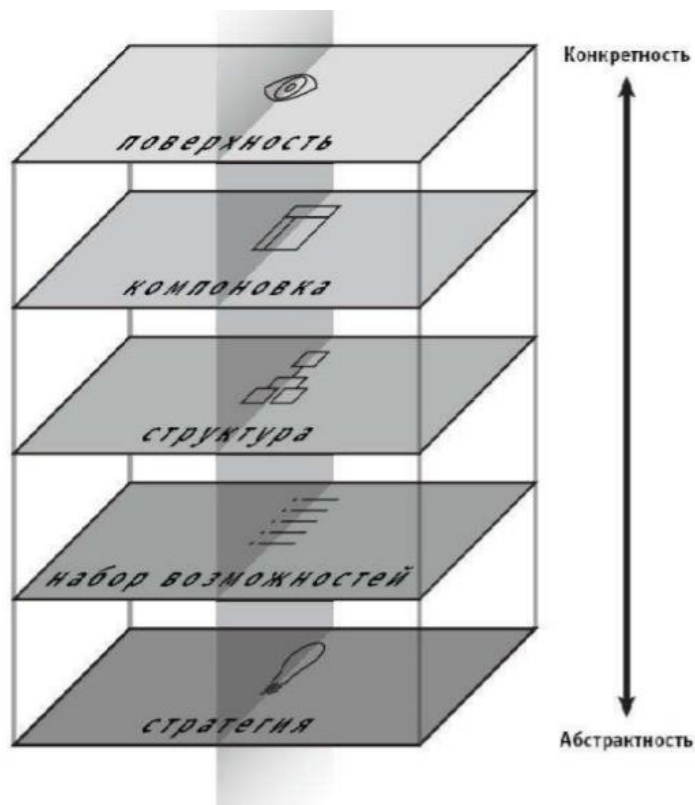


Рис. 5. Основные уровни проектирования взаимодействия

Начало работ над проектом

На этом этапе определяются объёмы работ, планируются затраты и т. п. Длительность этого этапа, как правило, не превышает 5–8% от общего времени разработки. Для адекватной оценки ресурсов (времени, денег, количества специалистов) требуемых для разработки (переработки) интерфейса, необходимо хорошо представлять себе объём информации, с которой следует ознакомиться. К ней относится информация о предметной области и прототипах. Она получается из литературных источников и опросов экспертов. Результатом этой работы является количественная оценка ресурсоёмкости проекта.

Чтобы предлагать адекватные интерфейсные решения, необходимо иметь ясное представление о предметной области системы. Предметная область изучается по литературе, кроме того, весьма полезны беседы с опытными пользователями, другими сотрудниками (экспертами) для выяснения всех деталей и характеристик предметной области. Вместе с «жалобами» заказчика на текущую версию системы результаты этого этапа составляют основное содержание работы над проектом (экспертная оценка часто обнаруживает проблемы, которые заказчику не видны, маскируясь под другие). Проблемы, выявленные на данном этапе, должны быть решены в новом интерфейсе. Удачные решения желательно сохранить, чтобы имеющимся пользователям не пришлось переучиваться (и чтобы сократить затраты на переделку). По сути, этот этап завершается созданием перечня удачных и неудачных интерфейсных решений (основное внимание уделяется решениям неудачным). На этом этапе проводится юзабилити тестирование текущей версии интерфейса, и составляются краткие протоколы и перечень выводов исследования.

Постановка задачи. Сбор информации о разрабатываемом продукте

На этой стадии анализируют данные о пользователях, формализуется функциональность и определяются критерии оценки проекта.

Залогом успешного проектирования пользовательского интерфейса является наличие наиболее полной информации об аудитории пользователей: их целях, задачах, предпочтениях, привычках и представлениях, и о заказчиках. Чем более полная информация

о продукте будет собрана и передана проектировщикам, тем более чёткое и правильное представление о его качествах будет у них сформировано и, соответственно, тем эффективнее будет проходить процесс разработки на всех последующих стадиях.

Данные о пользователях и о проекте должны содержать следующие позиции:

- характеристики пользователей: их опыт работы с компьютером, знание предметной области, мотивы, размер/важность групп пользователей, примеры (типовые ситуации) использования;
- цели и задачи пользователей;
- задачи проекта: что послужило причиной создания проекта, этапы создания проекта, какие результаты должны быть получены, какая информация необходима и когда;
- технология разработки и платформа, на которой будут работать пользователи;
- среда, в которой будет создаваться и использоваться проект (программная, физическая, рыночная, организационная и культурная).

Эта работа предполагает доступ к имеющимся и потенциальным пользователям системы, экспертам и проектной документации. На этом этапе разрабатываются пользовательские профили, модели пользователей. Обязательно должна присутствовать информация о субъективных ожиданиях пользователей системы. Без этого трудно или невозможно предугадать отношение пользователей к будущей системе. Поэтому должны быть описаны свойства, которым должен отвечать интерфейс для повышения субъективного удовлетворения, приведён перечень значимых для пользователей характеристик системы. Завершается эта часть работы описанием среды, в которой используется система, и основных характеристик ПИ. Характеристики ПИ отражаются в версии прототипа ПИ, которая на данном этапе будет, скорее всего, бумажной. Проводится юзабилити тестирование этой версии прототипа, и определяются скорость работы, количество человеческих ошибок, скорость обучения, субъективная удовлетворенность пользователей и т. д.

Иными словами, на данном этапе конкретизируются действительные цели проектирования нового интерфейса.

Необходимо чётко понимать, что пользователям не нужны инструменты сами по себе, нужны лишь результаты их работы. Никому не нужна программа обработки изображений — нужны уже обработанные изображения. Это значит, что пользователям просто нужно средство, делающее возможным выполнять определенную работу, достигнуть требуемого результата.

Формализация действий пользователей начинается с описания различных типовых сценариев. Для этого формализуются данные, необходимые пользователям для выполнения работы, последовательность самой работы, критерии завершённости этой работы. В результате должно появиться словесное описание взаимодействия пользователя с системой, не конкретизируя, как именно проходит взаимодействие, но уделяя возможно большее внимание всем целям пользователей. Количество сценариев может быть произвольным, главное, что они должны включать все типы задач, стоящих перед системой, и быть сколько-нибудь реалистичными. Сценарии очень удобно различать по имена существующих в них вымышленных персонажей.

Необходимо также на этом этапе проанализировать интерфейсы конкурирующих систем. Большая часть пользователей любой системы обладают навыками использования нескольких конкурирующих систем; если разрабатываемый интерфейс полностью отличен от конкурентов, то пользователям придется переучиваться. Кроме того, конкурирующие системы часто содержат эффективные решения, которые полезно перенять (или, по меньшей мере, учесть при проектировании интерфейса). Как и в случае экспертной оценки текущего интерфейса системы, отчёт по выполнению этого этапа работ содержит перечень удачных и неудачных интерфейсных решений, фокусируясь более на удачных решениях.

Рассмотрим основные способы сбора нужной, в процессе разработки пользовательского интерфейса, информации.

Исследование целевой аудитории

Оценивать результат проектирования в конечном итоге следует исходя из того, насколько успешно он отвечает требованиям пользователей и компании — инициатора разработки. Если у проектировщика нет ясного представления о пользователях, для которых выполняется проектирование, если у него отсутствует понимание имеющихся ограничений, организационных задач и бизнес целей, которые являются движущей силой разработки, то шансов на хороший результат очень мало — неважно, насколько при этом хороши навыки и творческие способности проектировщика.

Наиболее популярные методы сбора данных используют количественные методы. Эта группа методов даёт ответ на вопрос «Сколько?», а информация, получаемая в результате применения количественных методов, обрабатывается с использованием статистических методов анализа. Типичный результат использования количественных методов сбора информации — получение процентного распределения, какая часть выборки потребляет тот или иной продукт, знает данную марку, алгоритмы использования, свойства продукта и т. п.

В основе методик количественных исследований лежат чёткие математические и статистические модели, что позволяет в результате иметь не мнения и предположения, а точные количественные (число- 39 вые) значения изучаемых показателей. Особенность этой группы методов заключается в их высокой степени формализованности. Используемый инструментарий состоит из переменных, заданных исследователем заранее, он достаточно «жёсткий» и практически не меняется в рамках проекта. Высокая степень формализации количественных методик сочетается с их ориентацией на массовый сбор первичных данных и их статистическую обработку. При использовании количественных методов сбора информации исходной позицией является выборочная совокупность (выборка) и принцип репрезентативности.

Вместе с тем необходимо отметить, что настоящего понимания в изучаемых областях невозможно достичь, изучая только многочисленные отчёты, полученные в результате количественных исследований, хотя для ответов на другие вопросы эти данные могут оказаться крайне важными. Глубокое знание вопроса можно получить, лишь дополнив количественные данные качественными моделями и интерпретациями. Существует много видов качественных исследований, и каждое из них может сыграть важную роль в формировании общей картины проектируемого продукта.

Качественные исследования

Качественные методы сбора данных позволяют получить информацию, которая отвечает на вопрос «Почему?» и не может быть обработана при помощи статистических методов анализа. Иными словами, из информации, собранной с использованием какого-либо из качественных методов, нельзя вывести какие-либо проценты и распределения, она позволяет лишь понять различные факторы и мотивы определенных действий, выявить модели поведения и т. п. Качественные (неформализованные) методы ориентированы не на массовый сбор данных, а на достижение углубленного понимания исследуемых явлений. Отсутствие формализации делает невозможным массовый охват обследуемых объектов, в результате чего число единиц обследования часто снижаются до минимума. Отказ от широты охвата компенсируется «глубиной» исследования, т. е. детальным изучением явления в его целостности и непосредственной взаимосвязи с другими явлениями.

Качественные исследования позволяют получить глубокую, развернутую информацию о предмете исследования. В отличие от количественных, качественные исследования фокусируются не на статистических измерениях, а опираются на понимание,

объяснение и интерпретацию эмпирических данных и являются источником формирования гипотез и продуктивных идей. В качественных исследованиях широко используются проективные и стимулирующие техники — неструктурированные, не директивные способы задавать вопросы, которые помогают исследователю раскрыть мотивы, верования, установки, отношения, предпочтения, ценности, степень удовлетворенности, проблемы респондентов и прочее относительно разрабатываемого продукта или брендов. Проективные техники способствуют преодолению таких трудностей коммуникации, как вербализация чувств, отношений и т. п., а также выявлению латентных мотивов, неявных установок, вытесняемых чувств.

Качественные исследования помогают понять предметную область, контекст и ограничения продукта, более действенным способом, чем количественные исследования. Они также помогают выявить шаблоны поведения потенциальных пользователей продукта быстрее и проще по сравнению с количественными методами. Качественные методы позволяют изучить:

- поведение, взгляды, склонности потенциальных пользователей продукта;
- предметную область — технический, экологический, и деловой контексты разрабатываемого продукта;
- используемый лексикон и прочие социальные аспекты предметной области;
- способы применения существующих продуктов. Качественные исследования способствуют ходу проектирования, поскольку:
 - обеспечивают доверие и уважение к команде проектировщиков;
 - объединяют команду общим для всех пониманием особенностей предметной области и проблем пользователей;
- дают руководителям возможность принимать решения по тем или иным вопросам проектирования продукта на основе данных — вместо догадок и личных предпочтений.

Значение качественных исследований не ограничивается поддержкой процесса проектирования. Время, потраченное на подробное изучение пользовательской аудитории, способно принести плоды в виде важных для бизнеса решений, к которым невозможно прийти средствами традиционных исследований рынка.

Методы качественных исследований

Существует множество методов и приёмов проведения качественных исследований, но сосредоточим внимание на методиках, которые хорошо зарекомендовали себя в международной практике разработки пользовательских интерфейсов в течение последних десяти лет. Перечень основных методов качественных исследований:

- интервьюирование заинтересованных лиц;
- интервьюирование экспертов в предметной области (ЭПО);
- интервьюирование пользователей и покупателей;
- наблюдение за пользователями;
- обзор литературы;
- аудит продукта/прототипа и конкурирующих решений.

Интервьюирование заинтересованных лиц

Перед проектированием любого нового продукта должно проводиться исследование, с помощью которого получают представление о техническом окружении и бизнес контексте продукта. Практически всегда продукт проектируется для достижения одной или нескольких конкретных бизнес целей. Обязанность проектировщиков — создавать решения, не теряя из виду это бизнес цели, и поэтому крайне важно, чтобы команда проектировщиков начинала работу с изучения возможностей и ограничений, стоящих за краткой спецификацией проекта.

В общем случае заинтересованное лицо — это любой человек, обладающий полномочиями в отношении проектируемого продукта или несущий ответственность за какой-либо его аспект. Говоря более конкретно, заинтересованные лица — это ключевые члены организации, иницирующей работы по проекту; как правило, в этот круг входят высшие должностные лица, менеджеры и представители отделов разработки, производства, маркетинга, юзабилити, дизайна.

Интервьюирование заинтересованных лиц должно проводиться до начала любых исследований пользовательской аудитории, поскольку возникающие обсуждения нередко задают способы проведения пользовательских исследований. Обычно более эффективно интервьюировать заинтересованных лиц поодиночке, а не в группах, объединяющих несколько отделов.

От заинтересованных лиц важно получить информацию по следующим вопросам:

- предварительное видение продукта;
- бюджет и график проекта;
- технические возможности и ограничения;
- потребности бизнеса;
- представления заинтересованных лиц о пользователях.

Обсуждение этих тем важно для выработки общего языка и взаимопонимания между группами проектировщиков, руководителей и разработчиков. Работа проектировщика — создать видение продукта, в которое верит вся команда. Участники проекта вряд ли увидят в предлагаемых решениях отражение своей системы ценностей, если не потратить достаточное время на то, чтобы понять их взгляды.

Интервьюирование экспертов в предметной области (ЭПО)

На ранних стадиях проектирования неоценимый вклад даёт выявление и интервьюирование нескольких экспертов в предметной области — людей, сведущих в предметной области, на которую ориентирован проектируемый продукт. Как и заинтересованные лица, ЭПО способны представить продукт и его пользователей под интересным углом зрения, однако проектировщикам следует проявлять осторожность и понимать, что точка зрения ЭПО в определенном смысле искажена.

Интервьюирование покупателей

Покупатели — это лица, принимающие решение о приобретении продукта. Если речь идет о потребительских продуктах, покупатели зачастую являются пользователями продукта, хотя если продукты ориентированы на детей или подростков, то покупателями будут родители или другие взрослые, опекающие их. В случае с большинством технических, медицинских и производственных продуктов покупателем зачастую оказывается не конкретный пользователь, а кто-либо из руководства компании или соответствующий менеджер, имеющий свои цели и потребности, отличные от пользовательских. Чтобы сделать продукт жизнеспособным, важно понимать покупателей и их цели. Не менее важно осознавать, что покупатели редко сами пользуются продуктом, а когда все же делают это, то совсем не так, как пользователи.

В интервью с покупателями необходимо понять:

- каковы их цели в контексте приобретения продукта;
- что их не устраивает в существующих решениях;
- каков процесс принятия решений при покупке продуктов подобных проектируемому продукту.
- их роль в установке, обслуживании и управлении продуктом.
- проблемы предметной области и особенности используемой терминологии.

Интервьюирование пользователей

Пользователи продукта в процессе проектирования должны находиться в центре внимания. Именно эти люди лично пытаются добиться каких-либо результатов с помощью продукта. При перепроектировке или улучшении существующего продукта, важно общаться

не только с нынешними, но и с потенциальными пользователями, то есть с людьми, которые пока не пользуются продуктом, однако являются хорошими кандидатами на его использование в будущем, поскольку имеют потребности, удовлетворяемые продуктом, и входят в его целевую аудиторию. Интервьюирование обеих категорий позволяет выявить влияние, которое оказывает на поведение и образ мысли пользователя опыт работы с существующей версией продукта.

Проектировщики заинтересованы в том, чтобы получить от пользователей следующую информацию:

- контекст интеграции продукта в жизнь или рабочий процесс пользователей — когда, почему и каким образом применяется продукт;
- уровень осведомленности в предметной области с точки зрения пользователя — что необходимо знать пользователю, чтобы использовать предлагаемый продукт в процессе своей работы;
- существующие задачи и виды деятельности — как те, которые выполняются при помощи данного продукта, так и те, которые не поддерживаются им;
- цели и мотивы использования продукта;
- ментальная модель — как пользователи думают о своей работе и деятельности, а также чего они ожидают от продукта;
- проблемы и сложности при работе с продуктом.

Наблюдение за пользователями

Большинство людей не способны точно описать собственное поведение, особенно когда находятся в контексте своей деятельности. Также многие люди, из-за боязни показаться некомпетентными или невежливыми избегают обсуждать поведение программ, которое кажется им проблемным и непонятным. Из этого следует, что интервью, проводимое вне контекста ситуации, которую стремится понять проектировщик, даст менее полные и менее точные данные. Можно обсудить с пользователями их представление о собственном поведении, а можно непосредственно наблюдать это поведение. Второй вариант даёт лучшие результаты.

Для фиксации того, что говорят и делают пользователи, многие специалисты применяют технические средства, такие как аудио- и видеозапись. Следует проявлять осторожность и не пользоваться такими устройствами слишком бесцеремонно по отношению к пользователям, в противном случае они будут отвлекаться и вести себя не так, как в отсутствие записывающих устройств.

Обзор литературы

Параллельно с интервьюированием заинтересованных лиц команде проектировщиков следует изучить какую-либо литературу, касающуюся продукта или его предметной области. Сюда могут и должны быть включены маркетинговые планы, стратегия бренда, исследования рынка, опросы пользователей, технические спецификации и информационные материалы, статьи в деловых и технических журналах, связанных с предметной областью, сравнительный анализ конкурентных решений и т. п.

Команда проектировщиков должна собрать эту литературу и использовать её как основу для формирования списка вопросов к заинтересованным лицам и ЭПО, а затем в качестве источника дополнительных данных о предметной области и терминологии, а также для сравнения с уже собранными данными о пользователях.

Прочие виды исследований. Фокус-группы

В последнее время, особенно популярен метод сбора пользовательских данных посредством фокус-групп, в котором группа пользователей, отобранных из репрезентативной выборки по заранее выявленным демографическим параметрам, собирается в одной комнате и отвечает на структурированный набор вопросов или выбирает ответы из предложенных вариантов. Встреча записывается на диктофон или видеокамеру для последующего анализа. Фокус-группы — это стандартный приём в маркетинге

традиционных продуктов. Они полезны для оценки непосредственных реакций на форму продукта, его внешний вид или технический дизайн. Фокус-группы также позволяют оценивать реакции на продукт, которым респонденты уже пользовались в течение определенного времени.

Метод фокус-групп во многих случаях малопригоден в качестве инструмента проектирования. Фокус-группы хороши, когда нужно получить информацию о продуктах, которыми люди владеют или желали бы приобрести, но не очень подходят для сбора данных о том, что люди делают с этими продуктами, а также как и почему они это делают. Кроме того, фокус-группы в силу своей коллективной природы имеют свойство приходить к консенсусу: мнением группы становится мнение, высказанное большинством или озвученное громче других. Это в корне противоречит сути процесса проектирования, поскольку проектировщикам необходимо уловить все особенности поведения, чтобы учесть их при разработке продукта. Фокус-группы, как правило, выхолащивают то разнообразие поведения и мнений, которое как раз и должны приниматься в расчёт проектировщиками.

Высокоуровневое проектирование

Схематично работу на данном этапе можно представить, как показано на рис 6, в виде трёх взаимосвязанных блоков. Основной задачей разработки является преобразование потребностей потенциальных пользователей и целей конечного продукта в конкретные требования к контенту и функциональности разрабатываемого продукта.

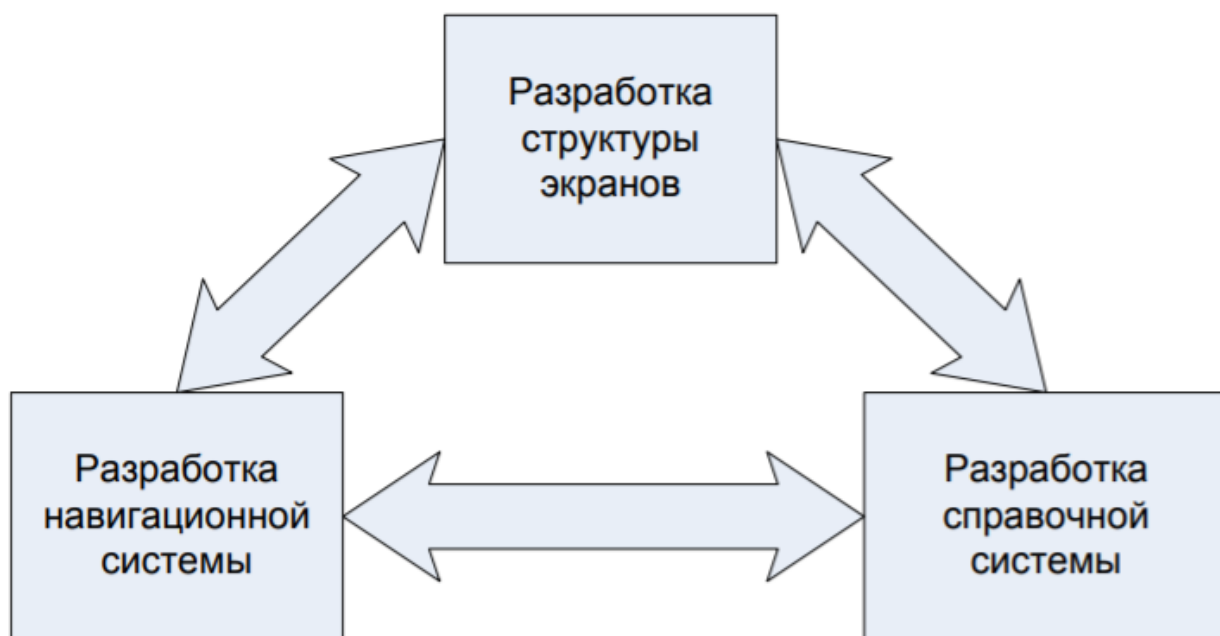


Рис. 6. Этапы высокоуровневого проектирования

Точное описание набора возможностей разрабатываемого объекта позволит всем участникам проекта сформировать более точное представление конечного результата, а также поможет эффективнее распределить обязанности между всеми участниками процесса разработки.

На уровне набора возможностей осуществляется переход от абстрактного вопроса: «Зачем мы делаем этот продукт?» к новому вопросу: «Что мы собираемся создать?»

На этапе разработки набора возможностей следует выделить два основных направления:

1. Функциональная спецификация. Документ, содержащий требования по функциональности разрабатываемого продукта.

2. Требования к контенту. Функциональная спецификация — документ, который включает в себе информацию о том, как будет работать продукт с точки зрения пользователя.

Функциональная спецификация обычно содержит в себе:

1. Сценарии пользователей.
2. Краткий обзор разрабатываемого продукта.
3. Набор возможностей продукта.
4. Перечень того, что не должен содержать продукт.

Необходимо помнить о том, что функциональная спецификация должна быть доступна всем участникам процесса разработки, и по мере продвижения процесса, она может корректироваться в связи с выявлением новых значимых возможностей и особенностей разрабатываемого продукта.

Первым шагом при создании функциональной спецификации является написание сценариев пользователей.

Сценарий пользователя — краткое, простое описание того, как пользователь пытается удовлетворить потребности с помощью разрабатываемого продукта. Представив процедуру, через которую могли бы пройти потенциальные пользователи, можно более точно выработать требования к программному продукту.

Сценарии создаются на основе информации, собранной в ходе определения проекта. Обычно в ходе интервью и непосредственного наблюдения за пользователями удается много узнать об их задачах. Цели стабильны и неизменны, задачи же неустойчивы, подвержены изменениям и часто оказываются ненужными в компьютеризованных системах. В процессе разработки сценариев следует находить и вычеркивать задачи, существование которых обусловлено лишь исторической необходимостью.

Эффективность сценария определяется в большей степени его охватом, чем глубиной. Иначе говоря, важнее, чтобы сценарий описывал процесс от начала до конца, чем чтобы он описывал каждый шаг в исчерпывающих подробностях.

Важно развивать лишь те сценарии, которые позволяют продвигаться вперед в процессе проектирования. Достаточно разработать лишь два вида сценариев, хотя сценариев каждого вида может быть и несколько.

На основе выявленных сценариев работы осуществляется разработка структуры экранов, т. е. определяется количество экранов, функциональность каждого из них, навигационные связи между ними, формируется структура меню и других навигационных элементов.

По сути, на этом этапе выделяются отдельные функциональные блоки. Под функциональными блоками будем понимать функцию или группу функций, связанных по назначению или области применения — в случае программы, и группу функций/фрагментов информационного наполнения — в случае сайта.

Существует три основных вида связи между блоками:

- логическая связь;
- связь по представлению пользователей;
- процессуальная связь.

Логическая связь определяет взаимодействие между фрагментами системы с точки зрения разработчика. Полученные связи очень существенно влияют на навигацию в пределах системы (особенно когда система многооконная). Поэтому, чтобы не перегружать интерфейс, стоит избегать как слишком уж отдельных блоков (их трудно найти), так и блоков, связанных с большим количеством других.

Связь по представлению пользователей важна, потому что пользователи имеют своё мнение о системе, и это мнение должно быть отражено в интерфейсе. Известно, что самый распространенный способ поиска, а именно поиск по классификации признаков, работает только в том случае, когда пользователи согласны с принципами этой классификации. Многие понятия не могут быть достаточно чётко классифицированы из-за большого количества значимых признаков. Кроме того, проблема состоит в том, что субъективный классификационный признак может отличаться от общепринятого. Из этой ситуации существует простой выход — способ карточной сортировки. Все понятия, которые требуется классифицировать, пишутся на карточках из расчета «одно понятие — одна карточка». После чего группе пользователей из целевой аудитории предлагается рассортировать эти карточки (при этом каждый субъект получает свой набор карточек). Получившиеся стопки из карточек нужно разобрать на составляющие и свести результаты от разных субъектов в один способ классификации.

Процессуальная связь описывает взаимодействие, не обязательно логичное, но естественное для процесса. Установление процессуальной связи обычно довольно трудная задача, поскольку единственным источником информации является наблюдение за пользователем. В то же время установление такой связи дело исключительно полезное. Жёстко заданная связь позволяет также уменьшить количество ошибок.

На основе разработанной структуры экранов на этом этапе выбирается наиболее адекватная навигационная система и разрабатывается её детальный интерфейс. Навигационная система показывает механизм распределения функций и задач между окнами программы. Навигационная система основывается на информационной архитектуре и связана с созданием организационных и навигационных схем, обеспечивающих экономичное и эффективное перемещение, как между различными задачами, так и внутри отдельной задачи. Информационная архитектура имеет прямое отношение к вопросам информационного поиска — проектированию систем, позволяющих пользователям легко находить нужную информацию. Однако архитектура web-сайтов, как отдельного вида информационных систем, часто призвана решать более широкие задачи, чем просто помощь в поиске информации: во многих случаях сайтам приходится обучать, информировать и убеждать пользователей.

Обычно решение задач информационной архитектуры требует создания классификационных схем, соответствующих целям разрабатываемого продукта, потребностям пользователей и его контенту. Существует два подхода к разработке такой классификации: нисходящий и восходящий.

Нисходящий подход к созданию информационной архитектуры заключается в её построении непосредственно на основе целей продукта и потребностей пользователей. Начиная с самых общих категорий будущего контента и функциональных возможностей, необходимых для достижения этих стратегических целей, проводится логическое разбиение категорий на подкатегории. Получившаяся иерархическая структура служит пустой оболочкой для контента и функциональности.

Восходящий подход к построению информационной архитектуры также состоит в выделении категорий и подкатегорий, но при этом в основу ложится анализ контента и функциональных требований. Начиная с имеющегося исходного материала, группируются элементы в категории низшего уровня, а эти категории — в более крупные, чтобы выстроить структуру, отражающую цели проектируемого продукта и потребности пользователей.

Для достижения наиболее эффективного результата целесообразно достичь баланса между нисходящим и восходящим подходами.

Структура разрабатываемого программного продукта, как и весь опыт взаимодействия в целом, опираются на понимание его целей и потребностей пользователей. Если изменились цели, которых собирались достичь разработчики с помощью данного ПО,

или потребности пользователей, которые мы стремились удовлетворить, следует быть готовым к соответствующей переработке структуры.

Организационные принципы, которым следуют на верхних уровнях, тесно связаны с целями разрабатываемого продукта и потребностями пользователей. На более низких уровнях архитектуры, применяемые организационные принципы в большей степени подвержены влиянию специфики контента и функциональности.

Определить основные разделы будущего продукта помогают такие способы исследования пользовательской аудитории как: анализ задач, создание персонажей и сценариев пользователей. Когда основные разделы определены, следующим важным этапом идет разработка иерархии и наполнение основных разделов проектируемого ПО.

Результаты данного этапа обязательно отражаются в презентационных, а возможно, и в псевдореальных версиях прототипа, по которым и проводится юзабилити-тестирование. В соответствии с результатами последнего осуществляются корректировки ПИ.

Интуитивно понятная навигация и положительные ощущения пользователя обусловлены не чем иным, как правильным расположением и представлением информации.

Структура справочной системы должна не просто описывать интерфейс, но и помогать пользователю решать его задачи. После разработки структуры экранов и навигации по ним составление соответствующей справочной системы обычно не вызывает трудностей.

Низкоуровневое проектирование

Низкоуровневое проектирование заключается в детальной проработке поставленных задач и в проверке качества разработанных решений (рис. 7).



Рис. 7. Этапы низкоуровневого проектирования

Концептуальная структура придает грубую форму массе требований, которые возникают из наших стратегических целей. На уровне компоновки проводится дальнейшее уточнение этой структуры, выделяя специфические аспекты дизайна интерфейса и навигации, а также информационного дизайна, которые сделают неосязаемую структуру вполне конкретной.

Этап разработки структуры определяет, как будет работать разрабатываемый продукт. На этапе компоновки определяется форма, которую примет эта функциональность. Кроме конкретизации представления информации, на этапе компоновки осуществляется переход к вопросам, принципиально требующим более глубокой детализации. Если на уровне структуры оперируют крупномасштабными понятиями архитектуры и взаимодействия, то на уровне компоновки рассмотрение сконцентрировано практически исключительно на отдельных страницах и составных частях.

При проектировании основных экранов производится полное описание их интерфейса (без обработки исключительных ситуаций), организация информации на экранах. К отчёту прилагаются макеты экранов с описаниями функциональности каждого интерфейсного элемента. Разрабатывается презентационный или псевдореальный прототип ПИ, а в конце этого этапа прототип вполне может быть и реальным. При юзабилити-тестировании на основе критериев оценки ПИ и сценариев действий пользователей

разрабатываются тестовые задания, которые выполняются пользователями на прототипе с фиксацией всех значимых характеристик деятельности (таких, как производительность 50 труда, количество человеческих ошибок). После этого выполняется подсчет соответствующих показателей и сравнение их с заданными (или желаемыми). На основании полученных данных интерфейс либо дорабатывается, либо считается разработанным. К второстепенным экранам относятся диалоговые окна и всевозможные сообщения. Их интерфейс полностью описывается, равно как описываются и исключительные ситуации, влияющие на интерфейс. При финальном юзабилити-тестировании на какой-то версии прототипа разрабатываются и выполняются тестовые задания, оставшиеся после предварительного тестирования. На основании полученных данных интерфейс либо дорабатывается, либо считается разработанным, т. е. это итерационная процедура (как и юзабилити-тестирование в целом).

Список использованных источников.

1. Баканов, А. С. Проектирование пользовательского интерфейса: эргономических подход / А. С. Баканов, А. А. Обознов. – М.: Изд-во «Институт психологии РАН», 2009.
2. Гаррет, Дж. Веб-дизайн: книга Джесса Гаррета. Элементы опыта взаимодействия / Дж. Гарретт. – СПб: Символ-Плюс, 2008.
3. Купер, А. Алан Купер об интерфейсе. Основы проектирования взаимодействия / А. Купер, Р. Рейман, Д. Кронин. Пер. с англ. – СПб.: Символ-Плюс, 2009.
4. Магазанник, В. Д. Человеко-компьютерное взаимодействие: Учебное пособие для вузов / В. Д. Магазанник, В. М. Львов. – Тверь: Триада, 2005.
5. Норманн, Д. А. Дизайн привычных вещей / Д. А. Норман. – М.: Вильямс, 2006.
6. Раскин, Д. Интерфейс: новые направления в проектировании компьютерных систем / Д. Раскин. – СПб.: Символ-Плюс, 2007.
7. Сергеев, С. Ф. Обучающие и профессиональные иммерсивные среды / С. Ф. Сергеев. – М.: Народное образование, 2009.

Виктор Олегович Федоров

**УПРАВЛЕНИЕ РАЗРАБОТКОЙ СИСТЕМ
АВТОМАТИЗИРОВАННОГО ПРОЕКТИРОВАНИЯ**

Конспект лекций