

УДК 004.7

В.Е. Силантьев

Харьковский национальный университет радиоэлектроники, Харьков

## ЭТАПЫ ПРОЕКТИРОВАНИЯ ПОЛЬЗОВАТЕЛЬСКОГО ИНТЕРФЕЙСА

*Интерфейс – система правил и средств, регламентирующая и обеспечивающая взаимодействие нескольких процессов или объектов. Пользовательский интерфейс (ПИ) – система правил и средств, регламентирующая и обеспечивающая взаимодействие программы с пользователем. Типичный ПИ имеет устройства ввода и вывода. Получив от пользователя команду, интерфейс «отвечает» ему, выводя разного рода информацию. Представление информации подразумевает выбор формата, носителя, структуры, композиции и визуальных приёмов. Исходя из стратегий разработки пользовательского интерфейса, можно выделить следующие этапы работы над UX-проектом. Перед началом работы над проектом формируются цели и задачи, следующим пунктом является анализ желаемой аудитории и целевой аудитории. На основе первых двух параметров прорабатываются варианты возможного взаимодействия с тем расчётом, чтобы максимально удовлетворить спрос. Уже исходя из этого, проектируется структура продукта и её внешний вид. UX / UI дизайн помогает сделать приложение комфортней для пользователя.*

**Ключевые слова:** интерфейс, юзабилити, прототипирование, дизайн, пользователь, продукт, сценарий взаимодействия.

### Введение

**Актуальность проблемы.** Проблема пользовательского интерфейса напрямую зависят от решаемых программным обеспечением задач, входных и выходных данных; однако при этом существует значительная свобода в том, в каком виде все эти данные будут представлены пользователю. От того, насколько пользовательский интерфейс будет функционален, понятен и удобен конечному пользователю, во многом зависит успешность решения поставленной при проектировании ПО. Хотя оценка качества пользовательского интерфейса процесс достаточно субъективный и трудно формализуемый, можно с уверенностью утверждать, что хороший интерфейс должен обеспечивать эффективную и производительную работу пользователя. Существует также и ряд критериев, которым должен удовлетворять качественный интерфейс.

**Цель статьи.** Разработка методики проектирования человеко-машинного пользовательского интерфейса, включая все этапы проектирования.

### Влияние юзабилити на пользовательский интерфейс

Юзабилити, как термин, пришел в оборот во время разработки первого человеко-ориентированного интерфейса и определяется как степень эффективности, удобства, продуктивности, при которой определенный пользователь решает определенные задачи в определенном контексте [3]. В веб-технологиях данный термин принято пони-

мать как присущую какому-либо проекту легкость использования. Легкость же использования выводится на основании конкретно взятого дизайна, навигации, архитектуры и прочих функциональных особенностей проекта. Многие ошибочно полагают, что "юзабилити" – это всего лишь простой и понятный интерфейс. Простота и понятность – это только один из аспектов технологических решений, которые используют веб-мастера и программисты для придания своим продуктам характеристик высококачественных показателей в плане юзабилити.

### Выбор графического дизайна при проектировании интерфейса

Одним из основных аспектов, на которых следует заострять внимание при создании хорошего юзабилити, – это дизайн. От того, насколько качественным будет дизайн вашего продукта, зависит очень многое. Именно удачные дизайнерские решения создают один из элементов хорошего юзабилити – профессиональный дизайн, который делает проект удобным для изучения, навигации и прочих действий пользователя. Дизайн создаваемого продукта должен отвечать определенным стандартам. Привычная схема располагает к большинству пользователей, как следствие, пользоваться таким шаблонным интерфейсом намного удобнее, нежели «навороченным» или же непривычным. В современном дизайне существуют для противоположных направления: скевоморфизм и плоский дизайн, имеющие свои достоинства и недостатки.

Скевоморфизм – это физический орнамент или элемент дизайна, который скопирован с формы другого объекта, но изготовлен из других материалов или иными методами. В качестве противоположности, обратной стороной монеты можно считать недавно ставший популярным «плоский стиль», главным примером которого является пользовательский интерфейс Microsoft Metro. Плоский стиль воплощает в себе визуальный минимализм, минимум текстур и эффектов подсветки для простых форм и плоских цветов. Однако, чтобы быть более точным, этот тренд не всегда касается скевоморфизма, который часто подразумевает связь с прошлыми воплощениями подобного дизайна, но довольно часто связан с реализмом: чисто визуальным стилем, который пытается имитировать реальные материалы и текстуры, продемонстрированным чрезмерным безвкусным использованием текстур в некоторых приложениях Apple. До появления iPhone реализм в пользовательском интерфейсе был чем-то необычным, за исключением видеоигр. С целью сохранить эффект погружения, разработчики игр давно начали делать пользовательские интерфейсы под дерево, металл и камень. Но в более «серьезном» контексте реализм все равно был ограничен. Единственным примечательным исключением были стеклянные кнопки «в стиле Web 2.0», унаследованные из более ранних версий Mac OS X.

Прежде всего, невозможно игнорировать визуальную привлекательность реализма. Сложные текстуры и анимация при превращении страниц сегодня кажутся скучными, после того, как последние пару лет мы пользуемся ими постоянно, но когда iPhone вышел впервые, никто до этого не видел такого визуального богатства в пользовательском интерфейсе системы (тем более, на телефоне).

Технические ограничения среды всегда будут влиять на искусство, которое среда может создавать, и это так же верно в отношении смартфонов, как и в отношении масляной краски. Действительно, существует несколько практических факторов, которые заставили реализм работать на iPhone. Также существует проблема в неверном понимании скевоморфизма: например, заставляя что-то выглядеть как физический объект, а не работать как он.

Для предсказания трендов в дизайне возникли целые отрасли в различных сферах. И, несмотря на это, никто не ожидал, что следующее движение в дизайне появится благодаря компании Microsoft.

Чтобы сильнее отличаться от Apple или потому, что они изначально считали это лучшей философией дизайна, Microsoft выбрали абсолютно

другую эстетику для своего пользовательского интерфейса в Metro-стиле. Пропали тени, подсветка, градиенты и текстуры iOS-приложений. Вместо этого Metro предлагает плоские цветные квадраты с крупным шрифтом.

Несмотря на то, обращает ли основная аудитория внимание на крупные блоки с плоским цветом или нет, новая философия дизайна Microsoft определенно взяла за душу техносферу, восхваляя особое внимание Metro-стиля к шрифтам и цветам.

Влияние мобильного веб-пространства на дизайн стало ощущаться еще в одном: отзывчивом дизайне.

Отзывчивый (или адаптивный) дизайн – это концепция, которая подразумевает, что вместо создания нескольких отдельных дизайнов для различных устройств, один и тот же дизайн должен адаптироваться под различные экраны. Он может быть в форме трансформации сетки из нескольких колонок в одну колонку, уменьшения больших шрифтов до более мелких, выпадающих меню, активируемых в ответ на касание к экрану телефона [2]. В первую очередь, как и любой другой тенденцией дизайна, им будут скорее злоупотреблять, так как многие дизайнеры сразу же перейдут на сторону большинства, не задумываясь о своем выборе. Но, если основной жертвой реализма был в основном хороший вкус, то чрезмерное использование минимализма может привести к серьезным последствиям в юзабилити.

Пользователи привыкли полагаться на большое количество тонких намеков, сопровождающих их через весь интерфейс: небольшой градиент на кнопках и скругленные углы, мягкие внутренние тени полей для заполнения, и «парящие» над остальным контентом панели навигации. Уберите все эти намеки, и у вас получится плоский мир, в котором все элементы внезапно оказались на одном том же уровне, сбивая вас с толку. Визуальный стиль – это не более чем средство достижения цели. Если ситуация требует реализма, развлекайтесь с текстурами и тенями. С другой стороны, если плоская эстетика лучше помогает дизайнеру достичь цели, тогда, наверное, стоит отказаться от градиентов.

## Определение и структурирование функциональности

Следующий шаг в начальном этапе проектирования интерфейса: структурирование контента. От того, какова будет архитектура предоставляемой информации, зависит многое. Темы, разделы, выпадающие меню – эти детали помогают пользователю ориентироваться, а также осуществлять навигацию.

При разработке пользовательского интерфейса анализ поведения потенциального покупателя, его действий и желаний является неотъемлемой частью начального этапа проектирования. Потенциальные покупатели (потенциальные клиенты) – все, имеющие возможность и способность купить товары или услуги ни разу не приобретавшие их, а также те, кто прекратил покупку товаров и(или) услуг по тем или иным причинам, но сохранившие возможность их приобретения. Для увеличения числа пользователей продукта следует обращать внимание на их поведение: на чем они задерживаются, каким маршрутом они перемещаются по навигации приложения, почему могут не доходить до целевых функций.

Для более подробного представления поведения пользователя следует обратить внимание на пользовательские сценарии.

### Пользовательские сценарии

Сценарии взаимодействия – это описание того, как должны работать функции системы. Они могут рассказывать о сути и особенностях работы функций, как в общем виде, так и в подробном, алгоритмическом. Первый вариант нужен для того, чтобы понять, зачем нужна и что делает функциональность. Второй по шагам расписывает все возможные сценарии использования продукта – что может сделать пользователь и как должна отреагировать на его действия система [1].

Диаграммы вариантов использования описывают функциональное назначение системы или то, что система должна делать. Разработка диаграммы преследует следующие цели: определить общие границы и контекст моделируемой предметной области; сформулировать общие требования к функциональному поведению проектируемой системы; разработать исходную концептуальную модель системы для ее последующей детализации в форме логических и физических моделей; подготовить исходную документацию для взаимодействия разработчиков системы с ее заказчиками и пользователями.

Суть диаграммы вариантов использования состоит в следующем. Проектируемая система представляется в виде множества сущностей или актеров, взаимодействующих с системой с помощью вариантов использования. При этом актером (actor) или действующим лицом называется любая сущность, взаимодействующая с системой извне. Это может быть человек, техническое устройство, программа или любая другая система, которая может служить источником воздействия на моделируемую систему так, как определит сам разработчик. Вариант использования служит для описания сервисов, которые система предоставляет

актеру. Диаграмма вариантов использования может дополняться пояснительным текстом, который раскрывает смысл или семантику. Примером пользовательского сценария может быть подробное описание последовательности действий определенной одной или нескольких функций, которые совершает пользователь для достижения определенных целей.

Целью составления пользовательского сценария является сбор функциональных требований. Сценарии взаимодействия позволяют не только перечислить функции системы, но и во всех подробностях рассказать о том, как они работают, оценить качество спроектированного интерфейса, составляющих его компонентов. Существует четыре основных критерия качества любого интерфейса, а именно:

1. Скорость работы пользователей.
2. Количество человеческих ошибок.
3. Скорость обучения.
4. Субъективное удовлетворение пользователей [4].

Скорость выполнения работы является важным критерием эффективности интерфейса. В чистом виде этот критерий ценят довольно редко, но почти всегда он является крайне желательной составляющей целого. Любая попытка как-то увеличить производительность труда всегда встречается положительно.

Длительность выполнения работы пользователем состоит из следующих составных частей: длительности восприятия исходной информации; длительности интеллектуальной работы (время осмысления пользователем, что он должен сделать); длительности физических действий пользователя; длительности реакции системы.

Длительность восприятия исходной информации особых комментариев не нуждается, пользователь должен представить себе, какая информация о выполняемой задаче у него существует, и в каком состоянии находятся средства, с помощью которых он будет решать данную задачу. Основное время здесь пойдет на считывание показаний системы.

Длительность интеллектуальной работы оценивается тем, что взаимодействие пользователя с системой (не только компьютерной) состоит из семи шагов:

1. Формирование цели действий.
2. Определение общей направленности действий.
3. Определение конкретных действий.
4. Выполнение действий.
5. Восприятие нового состояния системы.
6. Интерпретация состояния системы.
7. Оценка результата.

Из этого списка становится видно, что процесс размышления занимает почти все время, в течение которого пользователь работает с компьютером, во всяком случае, шесть из семи этапов полностью заняты умственной деятельностью. Соответственно, повышение скорости этих размышлений приводит к существенному улучшению скорости работы.

К сожалению, существенно повысить скорость собственно мышления пользователей невозможно. Тем не менее, уменьшить влияние факторов, усложняющих и, соответственно, замедляющих процесс мышления, вполне возможно. Например, применением метода непосредственного манипулирования и применения в интерфейсе эффективных методов при потере фокуса внимания. Потеря фокуса внимания приводит к потере пользователей, необходимо проектировать интерфейс так, чтобы пользователи возможно меньше о нем думали.

Для продолжения работы пользователь должен знать, на каком шаге он остановился; какие команды и параметры он уже дал системе; что именно он должен сделать на текущем шаге; куда было обращено его внимание на момент отвлечения.

Любое физическое действие, совершаемое с помощью мускулатуры, может быть или точным, или быстрым. Вместе точность и быстрота встречаются исключительно редко, поскольку для этого нужно выработать существенную степень автоматизма. Объясняется это физиологическими факторами: при резком движении невозможно быстро остановиться, соответственно, точное движение должно быть плавным и замедленным. Таким образом, чтобы физическое действие пользователя было быстрым, оно не должно быть точным. Пользователь, как правило, управляет компьютером двумя способами, а именно мышью и клавиатурой. Клавиатура не требует особой точности движений – неважно, быстро нажали клавишу или медленно, равно как сильно или слабо. Мышь, напротив, инерционна – есть разница между медленным её перемещением и быстрым, сильным приложенным усилием и слабым. Именно поэтому оптимизация использования мыши в системе может существенно повысить общую скорость работ. Возможные ошибки: ошибка определения действия; потеря цели; модальная ошибка.

Очевидно, что пользовательский интерфейс напрямую зависит от решаемых программными обеспечением задач, входных и выходных данных; однако при этом существуют значительная свобода в том, в каком виде все эти данные будут представлены пользователю. От того, насколько

пользовательский интерфейс будет функционален, понятен и удобен конечному пользователю, во многом зависит успешность решения поставленной при проектировании ПО.

## Прототипирование

На процесс проектирования пользовательского интерфейса немалое влияние оказывают субъективные представления проектировщика о понятности, удобстве и красоте. Поэтому, большое значение имеет проблема оценки качества пользовательского интерфейса. Проводя такие оценки на ранних этапах процесса проектирования, можно избежать большого числа ошибок, просчетов, неприятя ПО конечными пользователями.

Хотя оценка качества пользовательского интерфейса процесс достаточно субъективный и трудно формализуемый, можно с уверенностью утверждать, что хороший интерфейс должен обеспечивать эффективную и производительную работу пользователя. Существует также и ряд критериев, которым должен удовлетворять качественный интерфейс: лучше тот интерфейс, при котором время выполнения задачи меньше; лучше тот интерфейс, в котором число произвольных ошибок пользователя меньше, неоднозначность в понимании интерфейса должна быть минимальна (это способствует самообучению пользователей и делает их поведение предсказуемым); необходима высокая стандартизация интерфейса (она облегчает обучение пользователей); объем вводимой пользователем информации должен стремиться к минимуму (одни и те же данные не должны вводиться несколько раз), простота и визуальная привлекательность (удобство использования не менее важно, чем функциональность).

Прототипирование заключается в создании широкого набора макетов (прототипов) будущего пользовательского интерфейса, которые подвергаются сопоставительному анализу. Как правило, прототип содержит реализацию лишь самого интерфейса, без его функционального наполнения.

Цель прототипирования заключается в определении, насколько то или иное решение перспективно, и последующей реализации лучшего из возможного. Этот подход позволяет экономить время и ресурсы, затрачиваемые на проектирование и разработку.

Наиболее целесообразно применять этот подход на ранних этапах проектирования, что помогает выбрать правильное направление разработки, однако возможно и создание “локальных” прототипов для отдельных элементов пользовательского интерфейса. Таким образом, данный подход охватывает как проектирование интерфейса как целого, так и проектирование его частей.

Для создания прототипов привлекают не только специалистов, но и конечных пользователей, при этом полезны любые мнения, предложения и графические наброски; основная задача – создать 5 – 7 вариантов интерфейса, решающего одну и ту же задачу. При создании прототипов нужно исходить из разумного баланса между следующими ключевыми факторами: необходимый объем ресурсов для создания прототипа; планируемое время жизни прототипа (предназначен ли он для решения краткосрочной, локальной проблемы или для длительного, глубокого анализа); риск смены целей проектирования (сосредоточения внимания не на решении проблемы с помощью прототипов, а на создании самих прототипов).

Созданные прототипы подвергаются сопоставительному анализу, в связи с чем необходимо определить критерии оценки. Отправной точкой в определении таких критериев служит та проблема, ради решения которой были созданы прототипы. Ей может быть время ввода данных пользователем, время принятия пользователем решения на основе предоставленной информации, субъективная оценка качества интерфейса по некоторой шкале и т.п.

Проектирование интерфейсов на бумаге в 21 веке – самый быстрый, дешевый, кроссплатформенный и общедоступный метод. Значительная часть проектировщиков не использует бумажные прототипы и скетчинг по причине недоверия. Большинство специалистов ошибочно полагают, что такой простой и быстрый путь не поможет получить достаточно информации для работы над проектом. Жизнь вне монитора научила нас думать, что по-настоящему эффективные инструменты должны быть очень дорогими и расходовать много ресурсов.

Для проектирования интерфейсов появилось множество инструментов прототипирования. С их помощью можно получить хороший результат, но он не сопоставим со временем, потраченным на проработку и тестирование макетов. Буквально, три-четыре бумажных скетча главной страницы вполне достаточно для того, чтобы получить информацию о том, каким будет поведение пользователя и насколько интерфейс интуитивно понятен.

Бумажные прототипы значительно экономят бюджет. Существует гипотеза, доказать которую может каждый, попробовав в своей работе прототипирование на бумаге. Суть гипотезы в следующем: «Чем раньше мы получим информацию о юзабилити интерфейса, тем качественнее будет итоговый результат». Юзабилити разрабатываемой системы повысится, если на самой ранней

стадии будут выявлены и исправлены проблемы в интерфейсе.

Все знают, что тестирование юзабилити на каждом этапе проекта является обязательным, но чем раньше проблемы будут обнаружены, тем проще их можно вылечить. На этапе скетчей и набросков все еще можно, а в некоторых случаях даже нужно, делать фундаментальные изменения. Делать хорошо хотят многие, но по разным причинам не делают.

Целевой пользователь (Актёр) представляет собой любую внешнюю по отношению к моделируемой системе сущность, которая взаимодействует с системой и использует ее функциональные возможности для достижения определенных целей. При этом актёры служат для обозначения согласованного множества ролей, которые могут играть пользователи в процессе взаимодействия с проектируемой системой. Каждый актёр может рассматриваться как некая отдельная роль относительно конкретного варианта использования. Стандартным графическим обозначением актёра на диаграммах является фигурка человечка, под которой записывается имя актёра. В некоторых случаях актёр может обозначаться в виде прямоугольника класса с ключевым словом «актёр» и обычными составляющими элементами класса. Имена актёров должны записываться заглавными буквами и следовать рекомендациям использования имен для типов и классов модели. Примерами актёров могут быть: клиент банка, банковский служащий, продавец магазина, менеджер отдела продаж, пассажир авиарейса, водитель автомобиля, администратор гостиницы, сотовый телефон и другие сущности, имеющие отношение к концептуальной модели соответствующей предметной области.

Так как в общем случае актёр всегда находится вне системы, его внутренняя структура никак не определяется. Для актёра имеет значение только то, как он воспринимается со стороны системы. Актёры взаимодействуют с системой посредством обмена сообщениями с вариантами использования. Сообщение представляет собой запрос актёром определенного сервиса системы и получение этого сервиса. Это взаимодействие может быть выражено посредством ассоциаций между отдельными актёрами и вариантами использования или классами. Кроме этого, с актёрами могут быть связаны интерфейсы, которые определяют, каким образом другие элементы модели взаимодействуют с этими актёрами. Два и более актёра могут иметь общие свойства, то есть взаимодействовать с одним и тем же множеством вариантов использования одинаковым образом. Такая общность свойств и поведения представляется

в виде отношения обобщения с другим, возможно, абстрактным актером, который моделирует соответствующую общность ролей.

Структура интерфейса служит для спецификации параметров модели, которые видимы извне, без указания их внутренней структуры. В языке UML интерфейс является классификатором и характеризует только ограниченную часть поведения моделируемой сущности. Применительно к диаграммам вариантов использования, интерфейсы определяют совокупность операций, которые обеспечивают необходимый набор сервисов для актеров. Графический символ отдельного интерфейса соединяется на диаграмме сплошной линией или пунктирной линией со стрелкой с тем вариантом использования, который его поддерживает. Сплошная линия указывает, что связанный с интерфейсом вариант использования должен реализовывать все необходимые для него сервисы. Пунктирная линия со стрелкой означает, что вариант использования предназначен для спецификации только того сервиса, который необходим для реализации данного интерфейса.

## Выводы

Пользователь при обращении с интерфейсом должен представить себе, какая информация о выполняемой задаче у него существует, и в каком состоянии находятся средства, с помощью кото-

рых он будет решать данную задачу. Эффективность работы пользователя и его интерес обеспечивает правильно сформулированная методика разработки и проектирования пользовательского интерфейса.

Таким образом, интерфейс отделяет спецификацию операций системы от их реализации и определяет общие границы проектируемой системы.

## Список литературы

1. Раскин Дж. Интерфейс. Новые направления в проектировании компьютерных систем / Дж. Раскин. – М: Символ-плюс, 2004. – 272 с.
2. Гарретт Дж. Веб-дизайн. Элементы опыта взаимодействия / Дж. Гарретт. – М: Символ-плюс, 2008. – 192 с.
3. User experience как новый уровень качества программного обеспечения. Общие и комплексные проблемы естественных и точных наук [Электронный ресурс]. – Режим доступа к ресурсу: <http://cyberleninka.ru/article/n/user-experience-kak-novyy-uroven-kachestva-programmnogo-obespecheniya>.
4. What is an Interface? [Электронный ресурс]. – Режим доступа к ресурсу: <http://docs.oracle.com/javase/tutorial/java/concepts/interface.html>.

Поступила в редколлегию 13.10.2014

**Рецензент:** канд. техн. наук, проф. В.П. Ткаченко, Харьковский национальный университет радиоэлектроники, Харьков.

## ЕТАПИ ПРОЕКТУВАННЯ ІНТЕРФЕЙСУ КОРИСТУВАЧА

В.Є. Силантьєв

*Інтерфейс – система правил і засобів, яка регламентує і забезпечує взаємодію декількох процесів або об'єктів. Користувальницький інтерфейс (КІ) – система правил і засобів, яка регламентує і забезпечує взаємодію програми з користувачем. Типовий КІ має пристрої введення і виведення. Отримавши від користувача команду, інтерфейс «відповідає» йому, виводячи різного роду інформацію. Представлення інформації зважає на вибір формату, носія, структури, композиції та візуальних прийомів. Виходячи із стратегій розробки користувальницького інтерфейсу, можна виділити наступні етапи роботи над UX-проектом. Перед початком роботи над проектом формуються цілі і завдання з урахуванням всіх особливостей продукту. Наступним пунктом є аналіз бажаної аудиторії, яка матиме доступ до сайту. На основі перших двох параметрів опрацьовуються варіанти можливої взаємодії з тим розрахунком, щоб максимально задовольнити попит споживачів та завдання компанії. Виходячи з цього, проектується структура продукту і її зовнішній вигляд. UX / UI дизайн допомагає зробити додаток комфортнішим для користувача.*

**Ключові слова:** інтерфейс, юзабіліті, прототипування, дизайн, користувач, продукт, сценарій взаємодії.

## USER INTERFACE DEVELOPMENT STAGES

V. Ye. Sylantiev

*Interface is a system of rules and means of regulating and providing interaction of several processes or objects. User Interface (UI) – a system of rules and means of regulating and providing interaction with the user program. Typical UI has input and output devices. After receiving a command from the user, the interface is "responsible" to him, bringing all sorts of information. Presentation of information implies the choice of format, medium, structure, composition and visual techniques. Based on strategies for developing the user interface, the following stages of work on the UX-project. Before beginning work on the project, goals and objectives are formed taking into account all features of the product. The next point is the analysis of the desired target audience and that will have access to the site. Based on the first two parameters considering options possible interaction with the calculation to best meet the demand of consumers and the company's objectives. Already, on this basis, is projected structure of the product and its appearance. UX / UI design helps make the application more comfortable for the user.*

**Keywords:** interface, usability, prototyping, design, user, product, use case.