

Содержание

От автора	7
Предисловие	13
Как читать эту книгу	23

ЧАСТЬ I

Глава 1. Кто прочитает всё это?	27
Глава 2. Что за голос продукта?	34
Глава 3. А если автор видит иначе и хочет делать всё по-своему?	41
Глава 4. А нужен ли тут текст вообще?	47
Глава 5. Что, если всё ещё не нравится? Хорошо, а какие предложения?	55

ЧАСТЬ II

Глава 6. Без булшита и как будто для людей	66
Глава 7. Давайте по-честному	73
Глава 8. С учётом контекста и однородно	80
Глава 9. Компактно, понятно, полезно и грамотно... ..	91

ЧАСТЬ III

Глава 10.	Про химию форматов и формулировки для них.....	100
Глава 11.	Пишем кнопки.....	116
Глава 12.	Переключатели и радиокнопки тоже с текстом	125
Глава 13.	Чекбоксы — галочки в рамках.....	135
Глава 14.	Поле для ввода текста.....	142
Глава 15.	Как написать меню	149
Глава 16.	Просто текст — самый дикий.....	159
Глава 17.	Большая задача	170
Послесловие.....		181
Что ещё почитать, чтобы писать для интерфейсов лучше		183
И в чём писать		185

*Моей жене:
Красотка, я тебя люблю :-Р*

От автора

Как появилась идея всё это написать и зачем вам читать эту книгу

Привет, меня зовут Кирилл. Я профессионально пишу за деньги и около десяти лет делаю это для интерфейсов. Да-да, за тот текст в меню, на кнопках, в предупреждениях, формах обратной связи, сообщениях об ошибках и в прочих подобных штуках тоже кто-то отвечает.

Раньше такие тексты писали менеджеры и дизайнеры. В лучшем случае их проверял на опечатки и пунктуацию кто-то с хорошим русским языком. Затем эту обязанность «сгрузили» на копирайтеров и редакторов. Теперь есть специальный термин и целая новая профессия — UX-писатели.

Идея написать эту книгу появилась как ответ на очередной вопрос «Что почитать по UX-писательству?». Пожалуй, она могла бы и не выйти, если б не дизайнеры, с которыми я сейчас работаю в Сбере. Спасибо, ребята. Спасибо за вопросы про прикладной текст и всем остальным, кто их задавал, — в «Яндексе», «Иннове»

и «Айгайдсе», «Айфонсе» и «Аймобилке», где мне посчастливилось поработать. Ай, как много всего в России начинается на «Ай».

Отвечая на подобные вопросы, я много раз рекомендовал читать что-то о чистоте мыслей в голове и на письме — Корнея Чуковского, Нору Галь, Уильяма Зинсера. Все они хороши. Жаль, что не всегда это то, что надо. Никто из этих авторов не писал именно про текст в интерфейсе, хотя все ратовали за принципы, которых я сейчас придерживаюсь в своей работе.

Совсем недавно Максим Ильяхов собрал, осмыслил и модернизировал многое, о чём говорили в XX веке именитые писатели, редакторы и переводчики. Максим напомнил нам, что с тех пор ничего не изменилось. И это даже к лучшему: он, можно сказать, создал для многих пишущих людей хорошие рабочие места. На живых примерах Ильяхов показал владельцам продуктов, что платить писателю за каждую тысячу знаков — это даже не прошлый, а позапрошлый век. Текст не выйдет хорошим, если его цена растёт с объёмом. Цена растёт — ценность падает.

Илья Бирман, известный российский проектировщик, дизайнер и арт-директор, хорошо структурировал представление информации и нормально рассказывает про пользовательский интерфейс. Когда я пришёл к нему на интенсив в 2011 году, он спросил, зачем мне это нужно, ведь я копирайтер. Признаюсь, тогда у меня было лишь смутное представление о том, зачем мне это. Но я чувствовал, что так надо.

Уже тогда мне поступало много задач вроде «вычитать текст на лендинге, особенно кнопки и всё, что с ними непосредственно связано» и «глянуть скрипт для службы поддержки». Я смотрел то, о чём меня просили, и задавал странные вопросы: «А почему? Зачем? Для кого это? Как посетитель попал сюда и что будет после нажатия вот этой кнопки?» В современной рабочей среде такие вопросы — обычное дело. Но в начале 2010-х годов в большинстве случаев от копирайтера требовалось прочесть задание и выполнить всё, что в нём написано. В точности. А вопросы — для слабаков.

Уверен, Илья больше не спрашивает у своих студентов, зачем они пришли. Теперь всё настолько взаимосвязано, что невозможно чётко разделить процессы. Больше нельзя планировать по принципу «вначале мы только проектируем и рисуем дизайн, а слова вставим в самом конце». Хорошего результата не достичь, если выстраивать работу разных специалистов отдельно друг от друга.

Я решил написать эту книгу, чтобы упорядочить собственный опыт в UX-писательстве. Разложить всё по полочкам и для самого себя в том числе.

И, видимо, для вас, если вы её читаете. Для всех, кому интересно знать, как работать с UX-писателем или UX-писателем. Надеюсь, она поможет в обоих случаях. Я постараюсь рассказывать о своём опыте таким образом, чтобы было одинаково интересно начинающим редакторам интерфейсов и их нанимателям, дизайнерам, менеджерам — всем, кому приходится работать с писателями и с текстом в интерфейсе.

Хочется верить, что менеджерам книга поможет понять, зачем в команде UX-писатель и почему никто другой не должен брать на себя его обязанности. Мы же не заставляем иллюстраторов программировать, так почему же Фёдор из бухгалтерии должен вычитывать всё, что написали дизайнеры? Ну, пятёрка у него по русскому и врождённая грамотность, что с того?

Дизайнеры, скорее всего, найдут много общего между своей работой и работой редактора — поймут всю бессмысленность просьбы «написать текст вот сюда, потому что именно тут для него выделено определённое место». Или даже научатся сами писать с первого раза так, что редактору Виталине останется лишь «окнуть». Так тоже бывает.

Начинающим UX-писателям — бывшим журналистам, корректорам и людям прочих близких профессий — книга должна дать новые знания. А если нет, то хотя бы упорядочить те, что уже есть, и... помочь узнать что-то новое! Со мной такое много раз случалось — попытки всё упорядочить приводили к обретению знаний.

Надеюсь, результат этой моей работы подскажет и способы налаживания отношений в команде. Хочется верить, что из книги станет понятно, чем занимается и за что отвечает редактор, писатель, копирайтер, как бы вы ни называли пишущего человека в своей команде, и где его ответственность пересекается с ответственностью других профессионалов. Эта книга о том, как сделать всё хорошо и не переругаться в процессе.

Ну и никто не отменял других мнений. Всё изложенное дальше — описание только моего опыта. Если вы тоже давно и успешно пишете для интерфейсов, наши взгляды на профессию могут отличаться. И если вам есть что рассказать или о чём поспорить — сделайте это. Я постараюсь ответить. Возможно, ваша точка зрения окажется вернее моей, и в следующем издании книга выйдет исправленной и дополненной. Или вы найдёте время, чтобы выпустить собственную книгу, и сделаете всё лучше меня.

Я в самом деле надеюсь, что с утверждениями, приведёнными в книге, будут не только соглашаться, но она вызовет желание поговорить. И если молчать нет сил, напишите мне на ux@egegov.ru. Общаясь, мы отыщем истину и как-то поучаствуем в формировании новой профессии.

Предисловие

Что такое пользовательский опыт и при чём тут слова

«Да ну на фиг!» — сколько раз я, вы, все мы выкрикивали это или даже нечто грубее. Не понимали, как работает что-то новое, и бросали попытки разобраться в этом. Это непонимание возникает у нас при использовании многих цифровых продуктов — сайтов и мобильных приложений. Неприятные ситуации происходят и повторяются вообще всегда, когда мы как пользователи сталкиваемся с чуждой нам логикой и непривычными механиками.

Команда разработки может придумывать и делать удивительные вещи. И все их предположения могут оказаться ненужными нам, пользователям. Или востребованные функции будут реализованы настолько неудобно, что мы примемся вот так вскипать и бросать даже самые замечательные продукты.

И наоборот, мы как пользователи можем требовать такую функциональность и такие механики работы, которые кажутся непонятными и даже неприемлемыми команде разработки. Не в силах представить, что «вот это кому-то нужно», создатели сервисов, бывает, отказываются делать что-то действительно необходимое. Диалог не выстраивается из-за их нежелания понимать свою аудиторию, и это приводит к ухудшению пользовательского опыта. К ухудшению не дизайна, видимой оболочки, а самой сути — нарушает принципы работы вещей.

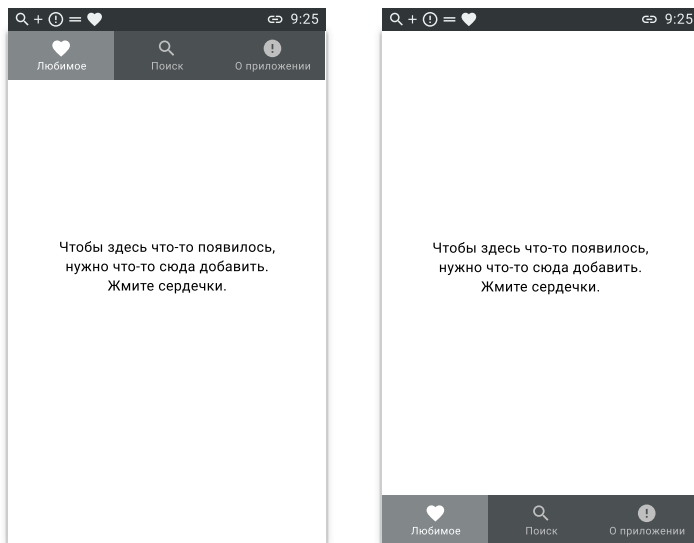
Стоп, что? Да, дизайн — это лишь видимая часть того, чем мы пользуемся. А пользовательский опыт — то, как мы это делаем. Например, нарисованная на экране кнопка — это в большей степени её дизайн. Форма, размер шрифта внутри неё, тень — всё это признаки видимого интерфейса, которые обычно описываются в дизайнерских гайдлайнах.

Или в дизайн-системах — как удобнее. Это такие многостраничные документы с описанием графических компонентов и правил их использования. Гайдлайны и системы помогают разрабатывать дизайн новых страниц быстрее и качественнее, а ещё с их помощью проще поддерживать единообразие в уже работающих продуктах.

Место же, где находится та кнопка, и принцип её работы — уже область пользовательского опыта. Если кнопка заметная и к ней удобно тянуться — пользовательский

опыт хороший, а если она спрятана, расположена неудобно — опыт плохой, нужно что-то менять. И вот тут разработчикам важно ориентироваться не на собственное видение или принятые в компании гайды, а на пользователей продукта: кто они, какие у них возможности и желания.

Сами посмотрите: ниже два совершенно одинаковых меню с одинаковым внешним видом — дизайном. Однако меню находятся в разных местах экрана — пользовательский опыт различается.



Чтобы дотянуться до меню на левом рисунке, нужно раза в два «растянуть» большой палец руки, в которой вы держите телефон, или задействовать вторую руку.

Меню в верхней части экрана унаследовано от старых устройств. Сначала такие элементы навигации скопировали из продуктов для компьютера на коммуникаторы со стилусами. Как на компе работает, так и на мобиле пускай будет. И это было нормально — пользователи всё равно управлялись с устройствами с помощью двух рук: в одной держали коммуникатор, а в другой — стилус, которым тыкали в экран. Потом эти верхние меню переехали как есть на смартфоны с ёмкостными экранами, которые воспринимают не точечное нажатие с усилием, а прикосновение пальца живого человека. И вот тут дизайнеры поняли, что на таких мобильных устройствах меню наверху — неудобно. Исправили это не сразу и не во всех продуктах — до сих пор встречаются «динозавры». Но в основном навигационные меню сейчас располагаются в нижней части экрана — как на примере справа. Вы можете убедиться в этом сами — посмотрите приложения на вашем смартфоне.

Всё это хорошо. Но всё же при чём тут слова?

Текст присутствует почти везде, где люди взаимодействуют с вашим продуктом. Даже больше того: когда пользователь сталкивается с чем-то непривычным или до этого ему неизвестным, именно внятное объяснение помогает ему не растеряться. И будь это объяснение текстовое или голосовое — не важно, лишь бы продукт говорил с человеком на понятном тому языке. Не значками или картинками, которые каждый новый

пользователь может понять по-своему, а нормально сформулированными фразами.

Бывает, люди даже читают всё, что написано на экране: пункты меню, объясняющие фрагменты текста, заголовки, надписи на кнопках, всплывающие подсказки, примеры заполнения форм, предупреждения, сообщения об ошибках и многое другое. Чаще всего подобное происходит при взаимодействии с совершенно новыми или значительно обновлёнными продуктами. Тут особенно важно, как вы всё сформулируете.

И это касается не только интерфейса самого продукта. Например, если речь идёт о новой версии мобильного приложения, то пользовательский опыт может транслироваться наружу — в тексте «Что нового» в магазине приложений. Можно сообщить об изменениях дежурно, сухо и без подробностей:

Исправили ошибки, устранили недочёты.

Но какие именно ошибки и недочёты? Изменили ли вы то, что так беспокоило меня весь последний год? А вот я помню, был серьёзный недочёт. Не помню, какой именно. Но точно помню, что был. Сейчас запущу приложение и сразу вспомню. Интересно, его вы тоже устранили?

Пользователям всегда лучше сообщать об обновлениях более развёрнуто и точно, ведь это всё сделано для людей. Например, можно выделить что-то конкретное:

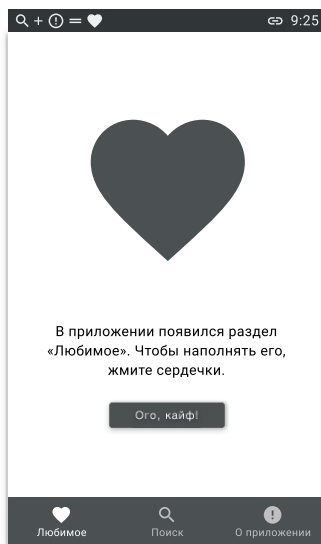
Помните, как приложение вылетало при редактировании профиля? Так вот, разработчики говорят, что устранили все возможные причины этого.

И дополнить общим:

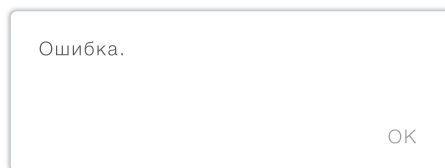
Заодно мы поправили разные мелочи. Попробуйте сделать всё то, что раньше вызывало ошибки. Если заработает как надо — хорошо. А если нет — напишите нам о недочётах на почту sapport@нашапочта.ру. Постараемся помочь.

Во втором случае текста намного больше, что как бы нехорошо. Но есть как минимум два аргумента в пользу такого варианта: в нём говорится о конкретных ошибках и он приглашает пользователей к диалогу, сообщает им о том, что разработчики тоже живые люди и понимают человеческую речь. А ещё второй вариант обещает пользователям не больше того, что действительно сделали разработчики. Он сообщает о конкретной правке и ещё нескольких менее значимых. А если осталось что-то ещё — пишите, вот почта.

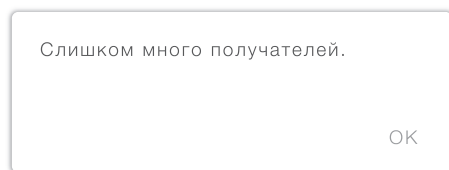
Идём дальше. Если в приложении появилось что-то полезное, не только исчезли ошибки, то при первом запуске сразу после обновления вы можете показать пользователям экран с рассказом о новинках. Или два, три экрана — насколько у вас хватит фантазии, а у пользователей — терпения:



Ладно, вот вы рассказали обо всех нововведениях. Или люди нажали на крестик в углу первого же сообщения и поспешили перейти к обычным сценариям использования приложения. Но вдруг что-то пошло не по сценарию и в приложении появилось сообщение об ошибке. Допустим, вы в любом случае увидите код ошибки в логах и поймёте её суть. А пользователю решили ничего не объяснять, для него написали максимально коротко и по делу:



Вроде ясно, что что-то пошло не так. Но непонятно, что именно, как исправить ситуацию, чтобы не увидеть это сообщение в следующий раз. Снова появляется работа для писателя. Он может пообщаться с командой, выяснить, допустим, что ошибка происходит из-за добавления слишком большого числа людей в получатели, и сообщить пользователю об ошибке примерно так:



Уже лучше. Человек понимает, как он может всё исправить прямо сейчас — хотя бы попытаться уменьшить число получателей. Но снова он остаётся отчасти в неведении. Пользователь не знает, что именно надо делать, чтобы больше не попадать в такие ситуации. Вынужден гадать с числами. И опять имеет лишь унылую кнопку тихого согласия с происходящим — «окей». Часто подобные ситуации можно улучшить, подсказав, что именно можно сделать сейчас и как поступать вообще. Писатель снова идёт общаться с разработчиками и выясняет новые подробности. Так у него получается лучшее сообщение, какое продукт может показать в этой ситуации:

Слишком много получателей

Пожалуйста, сократите список адресов хотя бы до 50.

ЛАДНО, Я ПОПРОБУЮ

Вот теперь наверняка ясно, что именно произошло и что делать, чтобы подобные сообщения больше не показывались.

Ещё раз взглянем на варианты. Первый, самый короткий, часто даже не пишут, а накидывают из-за отсутствия времени «на такие мелочи». Это скорее дежурная заглушка, которую иногда забывают заменить на что-то осмысленное, — так и уходит в релиз. Без обид и претензий — вариант разработчиков.

Второй вариант может написать копирайтер, которому как-то формулируют задачу — просят рассказать о том, что произошло. Такой текст пишется перед выпуском продукта, когда кто-то внезапно вспоминает: «Ой, у нас же там заглушка!»

Третий выдаст UX-писатель, который уже достаточно долго работает на проекте, привык разбираться в контексте и получать ответы на массу вопросов, прежде чем браться за формулирование.

Именно UX-писатели обычно пишут понятные и не раздражающие тексты для ошибок, кнопок, переключателей и всех остальных элементов интерфейса, участвуют в разработке голоса продукта и собирают гайды,

следят за единообразием и соблюдением требований типографики, часто берутся и за письма пользователям.

UX-писатель работает с дизайнерами, проектировщиками и исследователями с нуля — когда есть только концепция продукта и ещё нет внешнего вида, интерфейса. Такой специалист хоть и пишет, но это далеко не вся его работа — он сам немного дизайнер, исследователь, проектировщик и даже психолог. Он не просто «наваливает» текст в жёстко заданные рамки, а определяет эти рамки. Попутно указывает на сложности, которые могут возникнуть у пользователей, и расширяет «узкие горлышки».

UX-писатель выстраивает взаимодействие пользователя с продуктом в виде логично связанного диалога. Он постоянно улучшает качество этого диалога, сглаживает шероховатости и помогает людям проходить полезные сценарии — вот основные задачи такого специалиста.

Как читать эту книгу

Как многие нехудожественные издания, эту книгу можно читать почти в любой последовательности и даже не целиком.

Если вам надо только подсмотреть возможные формулировки интерфейсных элементов, то открывайте сразу третью часть. Там же больше всего картинок. В последних главах я на конкретных примерах показываю, как писать текст для кнопок, пунктов навигационных и контекстных меню, чекбоксов, переключателей, а также простой статичный текст.

Если вам нужно «только один хороший интерфейс написать», то будет достаточно третьей части. Полистайте её. После напишете всё максимально нейтрально по шаблону и сдадите проект, а дальше будь что будет.

Если вам захочется в самом деле понять, почему исходные примеры в третьей части плохие, зачем их переписывать и какими правилами можно руководствоваться в работе, начните читать книгу со второй части.

В ней всего четыре главы, и все про принципы, которые я обычно применяю. Это они, те принципы, заставляют меня и многих других UX-писателей писать определённым образом. Во второй части я отвечаю только на один вопрос: «Как?»

В первой части, где я постарался найти ответы на максимум вопросов, больше всего слов и, по мнению некоторых, меньше дела. В ней есть советы и для UX-писателей по работе в команде, и для руководителей, которым бывает трудно понять людей относительно новой профессии. Первые несколько глав я говорю об обычных трудностях в команде и способах их решения.

Если не спешить и не пропустить начало, то можно даже понять, как формировались мои рабочие принципы. Те, о которых я рассказал во второй части. А после можно не следовать моим или ещё чьим-то правилам, а дополнить их, изменить, доработать, создать собственные и аргументированно заявить о своём превосходстве.

Что пропускать, а что нет — решать вам. Вы, а не я, сейчас читаете эти строки, имеете определённые потребности и располагаете каким-то временем. Но я рекомендую вам прочесть эту книгу от начала и до конца. Люди, которых я просил прочесть её до публикации, сказали мне, что она хорошая.

Часть I

Глава 1

Кто прочитает всё это?

В начале работы над продуктом мы уже примерно понимаем его аудиторию — знаем всех, кто станет им пользоваться в первую очередь. Мы представляем себе определённые группы людей — им сколько-то лет, у них схожие привычки и доход, свой распорядок дня и ещё масса объединяющих группу параметров. Все наши будущие пользователи имеют выдуманные нами потребности, которые «закроет» наш продукт. С этим знанием у нас и появилась идея создать нечто новое.

Ну, или мы лишь примерно представляем себе, что нужно людям. Так тоже можно в самом начале. Мы примерно понимаем, что именно должны сделать, чтобы заполучить свою долю рынка, и этого уже достаточно.

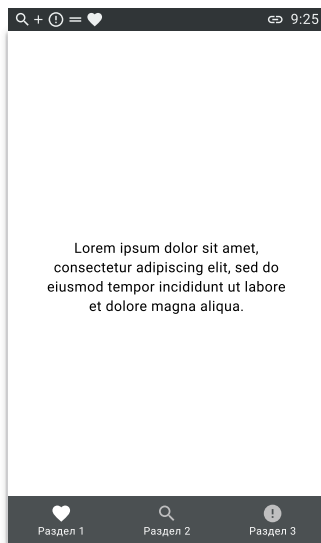
Составив список потребностей, мы принимаемся их исследовать. На этом этапе мы задаём все самые важные вопросы и понимаем, действительно ли стоит продолжать работу над задуманным. Мы выстраиваем процесс

именно так, потому что точно знаем, что это дешевле, чем работать вслепую и осознать нежизнеспособность продукта лишь на финальной стадии его создания — когда всё уже готово и потрачено много ресурсов.

Моделируя разные ситуации, мы тестируем гипотезы и выясняем, что действительно нужно включить в первую версию продукта, а на что не стоит тратить времени, что можно оставить на потом. Выясняем, чем мы «зацепим» людей. Исследуем, скрупулёзно документируем, сопоставляем и выбираем лучший из возможных путей.

Завершаем этап проектирования пользовательского опыта и начинаем «накидывать» реальный внешний вид продукта. Допустим, мы задумали выпустить мобильное приложение. У него появляются свои цвета и формы. Логика, о которой раньше мы только думали и которую закладывали в прототипы, обретает чёткие очертания.

Наступает тот момент, когда проект уже должен показать себя, — пора его представить людям. И тут мы понимаем, что не можем собрать приличную версию. Большинство текстовых блоков, сообщений об ошибках, подсказок и кнопок заполнено обычными «рыбными» текстами:



Хорошо, предположим, тексты не совсем «рыбные» — не дизайнерские заглушки на латыни. И они даже нормально работали в прототипах на фокус-группах. Только вот их стыдно показать людям — реальным пользователям, которые получают продукт не бесплатно, а им придётся за него заплатить. Или заплатить за какие-то расширенные возможности. Или потратить время на просмотр рекламы. Даже зарегистрироваться — уже напряг. Так или иначе, их вход будет сложнее, чем у людей, которые участвовали в ваших экспериментах и получили за это деньги. И вот тут нужен особенный подход.

Хорошо, если мы понимаем, что успех продукта зависит в том числе и от его речи — от того, как он говорит

с пользователями. Так мы приходим к мысли, что пора нанять копирайтера. Сказано — сделано. Вот он уже в команде. И он ноет! Тут в интерфейсе ему места мало, там с логикой что-то не так. Здесь вообще, говорит, не нужна кнопка и текст не нужен. Или не нужно столько текста, сколько на него заложили места в прототипе. Начинает подвывать дизайнер. Разработчики, которые уже залили тексты прямо в код, тоже недовольны — им теперь приходится отыскивать «плохие» строки и менять их на «хорошие». А чем «плохие» плохи, если и с ними было неплохо?

Возникает впечатление, что не того человека писать тексты наняли. На собеседовании вроде нормальный копирайтер был, адекватный, а в команде сразу же повёл себя как капризный ребёнок. И если его сейчас уволить, то кто доделает работу? Новый придёт и тоже начнёт ныть, решит переписывать уже переписанное. Они же как программисты — каждый новый ругает предшественника и говорит, что раньше всё делали не так. Вот он-то — воплощение опыта и всего прекрасного. Он возьмёт ещё полгода и всё сделает как надо. А у нас уже нет половины года. Так продукт никогда не выйдет!

Может, просто первого поздно наняли? Точно! Вот в чём дело. Заменять текстовые заглушки в интерфейсе красивыми осмысленными формулировками в последний момент — одна из грубейших ошибок в разработке продуктов. И что же мы раньше об этом не подумали?

Смотрите, что происходит, когда копирайтера или редактора просят вписать осмысленный текст в уже или

почти свёрстанный продукт. Всё начинается с заглушек с шаблонным описанием:

Определять местоположение

- ☐ Lorem ipsum
- ☒ Lorem ipsum
- ☐ Lorem ipsum
- ☐ Lorem ipsum

Дальше появляется смысл, но он никак не лезет в установленное кем-то ограничение:

Определять местоположение

- ☐ всегда
- ☒ когда приложение запущено
- ☐ раз в 1 час
- ☐ никогда

И тут вроде надо расширить блок:

Определять местоположение

☐ всегда

☒ когда приложение запущено

☐ раз в 1 час

☐ никогда

Но дизайнеры против, у них из-за этого всё разъезжается и выглядит некрасиво. И вариант написать в две строки им тоже не нравится. В таком случае ширина блока остаётся прежней, но увеличивается высота:

Определять местоположение

☐ всегда

☒ когда приложение запущено

☐ раз в 1 час

☐ никогда

В итоге страдают все. Много лишней работы появляется не только у редактора, а ещё и у дизайнера и фронтендера. Руководитель группы разработки тоже не сидит на месте и тратит время на переговоры и организацию команды.

Вот рынок мы изучили, персонажей-пользователей собрали по кусочкам — как из деталей конструктора,

продумали вроде нормальную логику работы приложения, протестировали прототипы, разработали дизайн и всё запрограммировали. Там процесс, здесь процесс. Всё по науке. Если не считать тексты, которые оставили на потом.

Привлекать писателя только в конце разработки — всё равно что впервые учить человека публично говорить в 22 года. Ему, судя по возрасту, пора защитить диплом, выпускиться из вуза и устроиться на работу, а он, простите, ни бе ни ме.

Не умея говорить и хотя бы рассказывать о себе, молодой специалист устраивается работать в лучшем случае стажёром. А чаще соглашается на место, никак не связанное с только что полученным дипломом. И онлайн-интенсив ораторства, который недавний студент прошёл перед защитой, вообще не помогает.

Обычная и довольно распространённая история — когда недавние выпускники понимают, что пять лет в университете не дали коммуникативных навыков, а с эмоциональным интеллектом и вовсе провал.

С продуктами всё так же. Если они не говорят со своими пользователями или говорят неумело, люди либо не подозревают о каких-то их возможностях, либо считают их использование необоснованно запутанным. В обоих случаях у продукта отсутствует голос, а сам продукт выглядит чем-то непонятным и кажется сложным. В итоге пользователи восклицают: «Фигня какая-то!» — и находят замену.

Глава 2

Что за голос продукта?

Мы — люди и общаемся друг с другом на человеческих языках. Их очень много, в мире насчитывается более 7000 диалектов. Наиболее распространённых, на которых говорит 4/5 населения планеты, — около 80. В России говорят на 37 языках, в том числе на очень редких. Государственный язык, который понимает большинство жителей страны, один.

И если вы выпускаете продукт в России, он точно должен уметь прилично говорить на русском. То есть не так:

Внимание!

Ошибка обработки запроса
BillingPayPrepRq В ЦПФЛ! Нет
начислений

ОК

Тут стоит задать сразу несколько вопросов разработчикам или менеджеру, который решил показать это сообщение пользователям. Вот что приходит на ум в первую очередь:

Что за запрос такой и почему может возникать ошибка?

Что такое ЦПФЛ?

Что можно сделать в этой ситуации, кроме как «окнуть»?

Зачем так орать на пользователя?

Не будем сейчас искать ответы на эти вопросы. Тем более что просто переписать сообщение по-русски обычно всё равно мало — даже на одном языке можно говорить совершенно по-разному. Ну, примерно на одном языке. Помните, как вы агрились, потому что вас не понимали взрослые? Всё приходилось им разжёвывать, как маленьким. Или, наоборот, как вы объясняли своему ребёнку, что такое лаконичность. Ну или ещё какое-нибудь мудрёное понятие. Вы искали все новые слова, подбирали выражения вроде понятные, но на деле — ничего подобного. Всё работало против вас, и каждое новое использованное слово тоже приходилось объяснять, чтобы прояснить значение того первого. В итоге вроде простой вопрос выливался в утомительную лекцию.

Чтобы с вашим продуктом не возникало таких проблем, чтобы не приходилось объяснять каждый экран, мало просто расшифровать ЦПФЛ — всё равно не поймут.

Продукт должен уметь общаться на простом современном русском языке. Это вам скажет любой человек, который хотя бы немного понимает в текстах.

Поэтому когда я прихожу на новый проект, то сначала стараюсь избавить его речь от всего непонятного и сделать её максимально простой, чтобы уяснить всё самому. Долой жаргонизмы, аббревиатуры и всё подобное. Для понимания какого-то слова нужен словарь? В топку это слово, и словарь туда же. Нужна сноска — слово или целое предложение летит следом. Ничего не понятно, а пальцы уже сами тянутся открыть новую вкладку браузера со статьёй в «Википедии»? Ищем простую и ясную замену.

Смотрите, что происходит, если оставить в интерфейсе непонятные аббревиатуры:

Переходите на новое приложение

Скоро МП «Мой кошелёк» будет отключён.

[СКАЧАТЬ НОВОЕ](#)

Не разобравшись с тем, что означает МП, автор текста присвоил ему мужской род. Что? Кошелёк. С ним что будет? Он будет отключён. А, ну так и напишем. Но если понять смысл коварной аббревиатуры МП, то оказывается, что всё надо переписать. МП — это мобильное приложение. Оно. И вернее говорить, что оно будет отключено.