

# Интерком о продакт менеджменте

Перевод книги [Intercom on Product Management](#)



**Некоторые полезные мысли о трудностях создания программных продуктов от команды Интерком. Наш продукт помогает компаниям видеть, кто пользуется их приложением и легко общаться с пользователями через email или прямо в приложении.**

<https://www.intercom.com/>

Мы также регулярно делимся в нашем блоге мыслями о продакт менеджменте, дизайне, стартапах и бизнесе.

<https://www.intercom.com/blog/>

Иллюстрации обложки и в главах: Intercom Brand Studio

Диаграмма привычек в главе 1: Нир Эяль, автор книги “На крючке”

Блог: NirAndFar.com

© 2017 Intercom Inc.

Мы избавим вас от юридических мамбо-джамбо. Но пожалуйста, если будете делиться книгой или ее частями, упоминайте авторов и оставляйте соответствующие ссылки.

Перевод на русский: Алексей Горобчук [aliaksei.harabchuk@gmail.com](mailto:aliaksei.harabchuk@gmail.com)



# Оглавление

[Введение](#)

[Глава 1. Анализ текущего состояния вашего продукта](#)

[Глава 2. Когда говорить «нет» новому функционалу](#)

[Глава 3. Какие функции добавлять в продукт](#)

[Глава 4. Что делать, чтобы функцией пользовались](#)

# Введение

В Интерком мы верим, что в каждом стартапе основной фокус должен быть на выдающемся продукте. Продакт менеджер — роль, в которой есть элементы инженерии, маркетинга, исследований и управления проектами, и такая роль является ключевой в создании выдающихся продуктов. Она требует непрерывного внимания к целостности продукта, объединяя под единым видением то как продукт проектируется, создается, называется, брендируется и продвигается на рынке.

Эта дисциплина все еще молодая и слабо определена из-за постоянно меняющейся природы разработки программных продуктов. Многие считают, что это даже не отдельная дисциплина в наши дни. В мире стартапов иногда становятся продакт менеджерами не осознанно, а потому что кому-то надо выполнять эту роль.

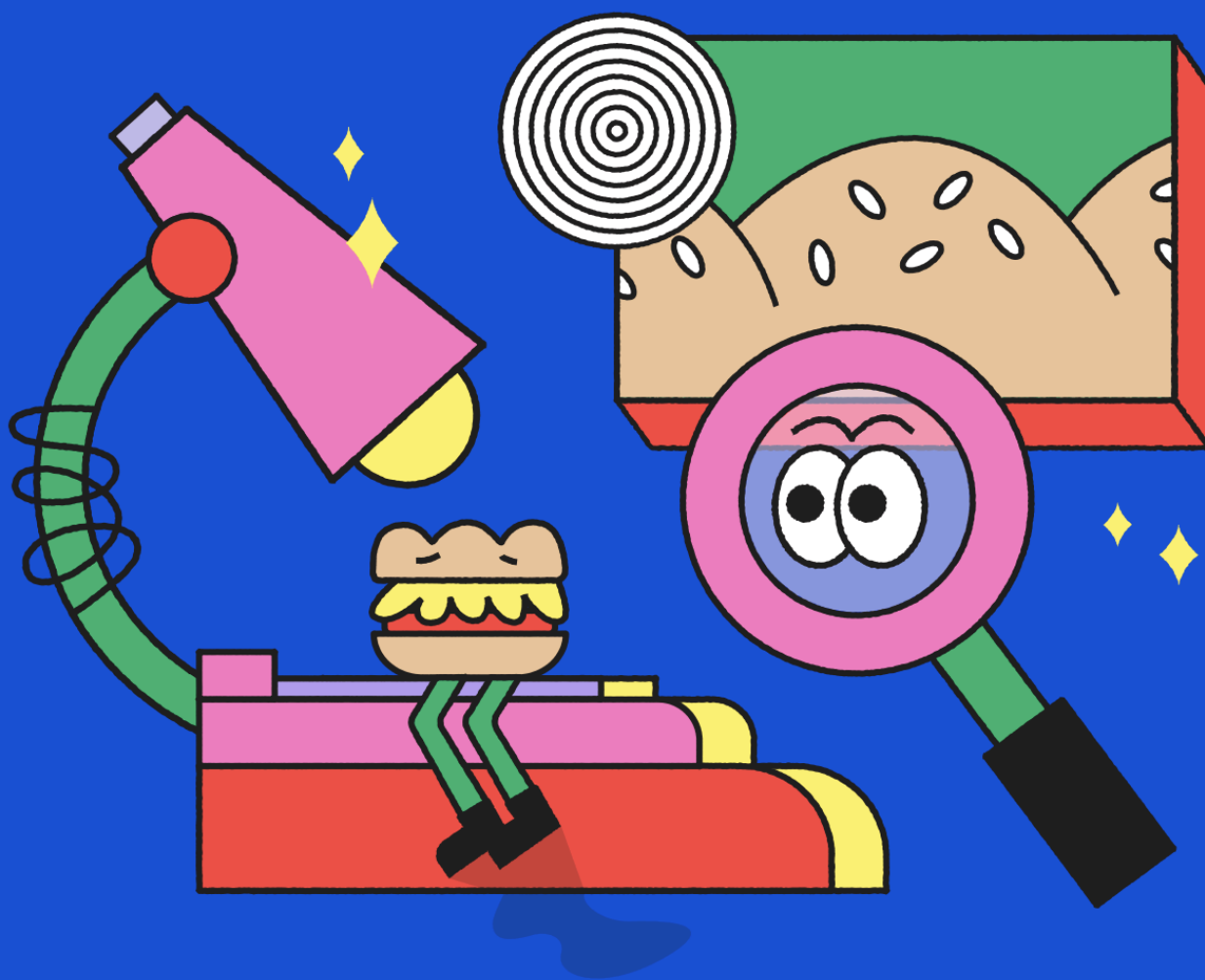
Независимо от того, каким путем вы пришли, или какая ваша текущая роль, если вы работаете над продуктом - мы создали эту книгу для вас.

Мы пишем в нашем блоге «[Внутри Интерком](#)» об уроках и опыте разработки продукта более шести лет. Эта небольшая книга связывает наиболее релевантные статьи из блога о разработке и управлении продуктом. Мы надеемся, она поможет ускорить ваше развитие в этой области. Постоянные читатели блога заметят, что мы добавили несколько разделов, чтобы связать идеи разных статей.

Мы начнем с оценки текущего состояния вашего продукта. Посмотрим какие части используются и какие требуют улучшения. Во второй главе вы узнаете о трудном выборе, который возникает при добавлении нового функционала, и почему «нет» - это самое важное слово в словаре продакт менеджера. Третья глава содержит чек лист, который должен пройти каждая новая функция; в

этой же главе обсудим, как выпускать новый функционал. В завершение вы узнаете, как помочь пользователям начать использовать новые функции. Потому что конечная цель выпуска нового кода, чтобы он использовался и был полезным.

**Поехали!**



## Глава 1. Анализ текущего состояния вашего продукта




Начнем сначала. Вы продакт менеджер? Основатель стартапа? СЕО? СТО? Как бы ни называлась ваша должность, предположим, вы принимаете решения о вашем продукте. Примерно такие: какой функционал разрабатывать, когда это сделать, как помочь пользователям начать пользоваться им. У нас у всех большие надежды и мечты о наших продуктах и как они будут жить, когда попадут в большой жестокий мир. У вас, возможно, блокноты полные идей, как сделать следующий Uber, Slack или Facebook. Но прежде чем сделать следующий шаг на пути к великому, нужно подвести итоги и посмотреть, что происходит с вашим продуктом сейчас.

Что на самом деле пользователи делают с вашим продуктом?

Все ли пользователи используют все доступные функции? Конечно, нет. Давайте разберемся с этим.

Когда вы планируете развитие продукта, и на что ваша команда потратит время, полезно спросить себя «сколько людей используют каждую из функций нашего продукта?». Это основы продакт менеджмента и, возможно, потребует некоторого времени, делая запросы к базе, анализа статистики или несколько секунд, чтобы узнать это в Интерком. Простой способ визуально представить использование функций - это построить таблицу в двух измерениях, сколько человек используют функцию, и как часто.

У вас получится что-то похожее на такую таблицу.

Все время	 Calendar	Track Time	Docs Editor	 Messages
Часто	 Chatroom		Feature X	Uploads
Иногда				Feature Y
Редко				
	Несколько пользователей	Многие пользователи	Большинство пользователей	Все пользователи

Типичный аудит функционала

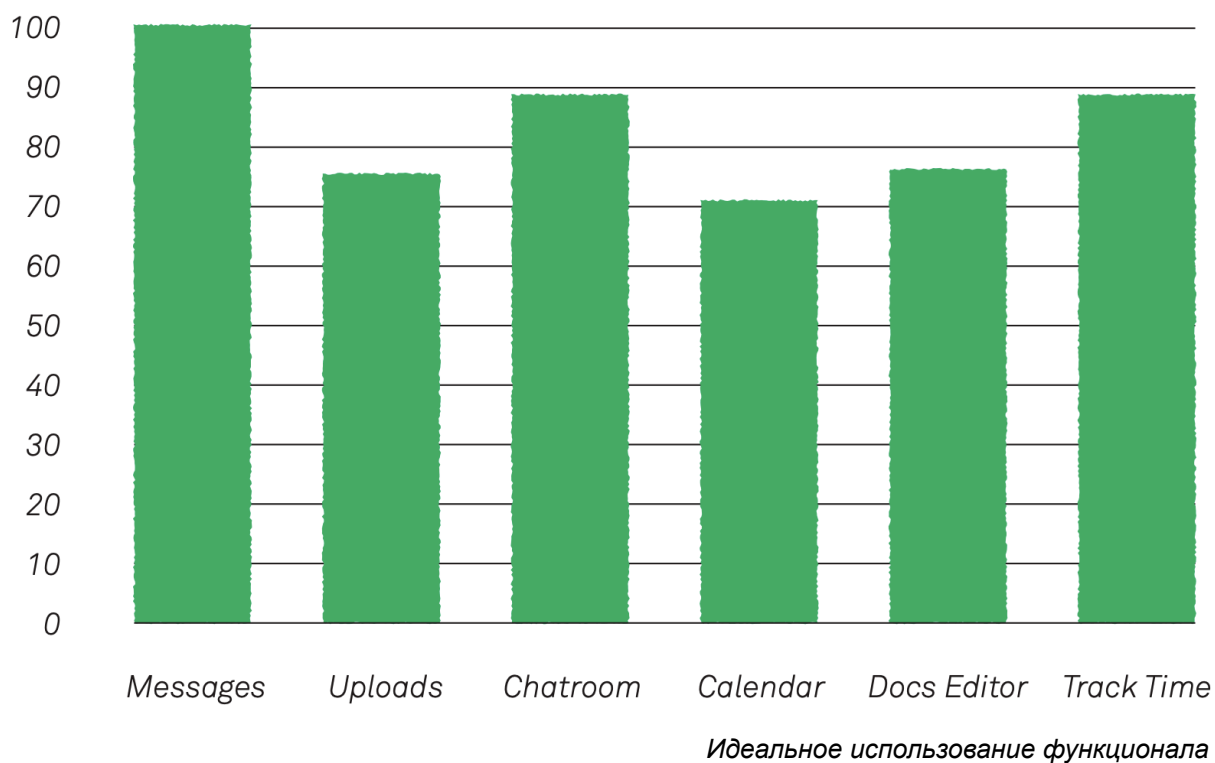
Основная ценность вашего продукта в правом верхнем углу, там где звезда, потому что это тот функционал, которым пользуются.

**Примечание:** Исключите административные функции, такие как создание учетной записи, сброс пароля, и т.п. из этой таблицы. Также исключите функционал, предназначенный только для некоторых пользователей, например, корпоративных пользователей. Такой функционал нужно анализировать отдельно.

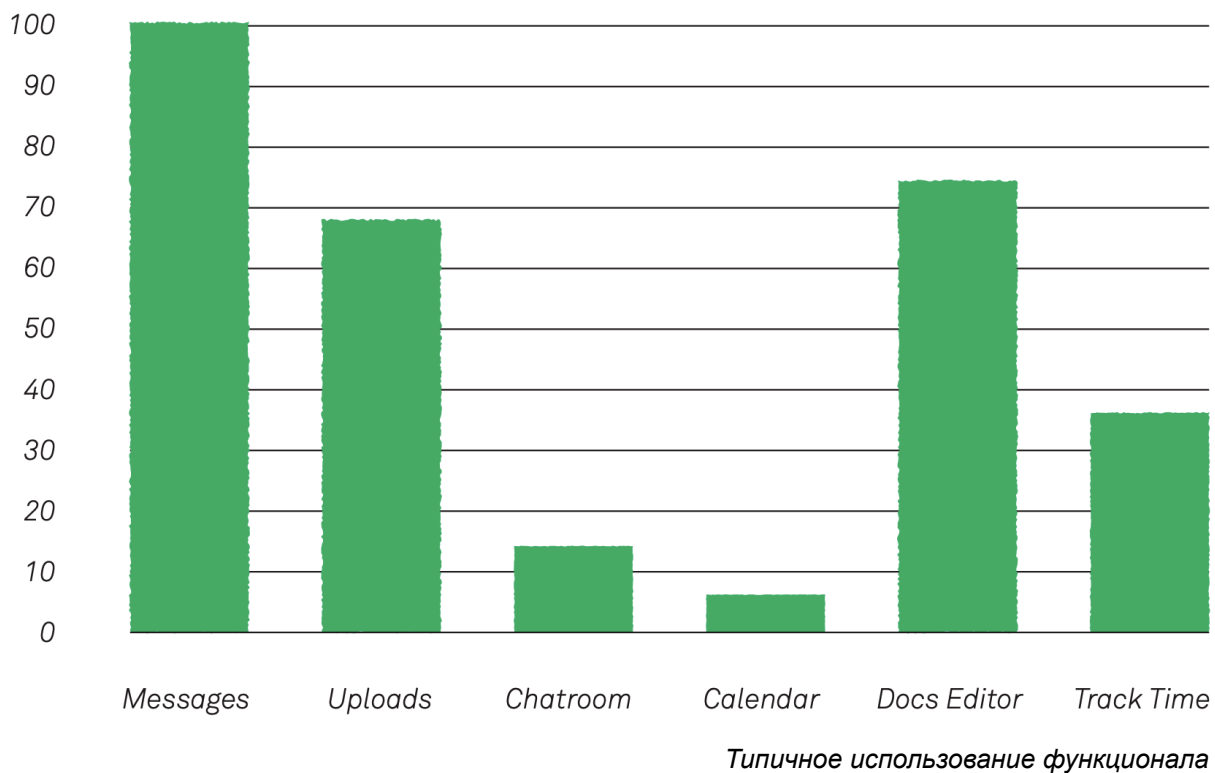
Если у вас есть функции в левом верхнем углу, это значит мало пользователей приняли их. Другими словами, мало пользователей зависят от них, а большинство крайне редко используют их.

Еще один вариант анализа: увидеть какой процент ваших клиентов приняли каждую функцию. Вы можете представить это в виде простой диаграммы. На диаграмме ниже показан идеальный продукт. Тот самый, о котором мы думаем, когда рисуем его в Photoshop. Все мои пользователи будут использовать все функции, верно?





Но как каждый продакт менеджер обнаруживает позже, что суровая реальность выглядит так:



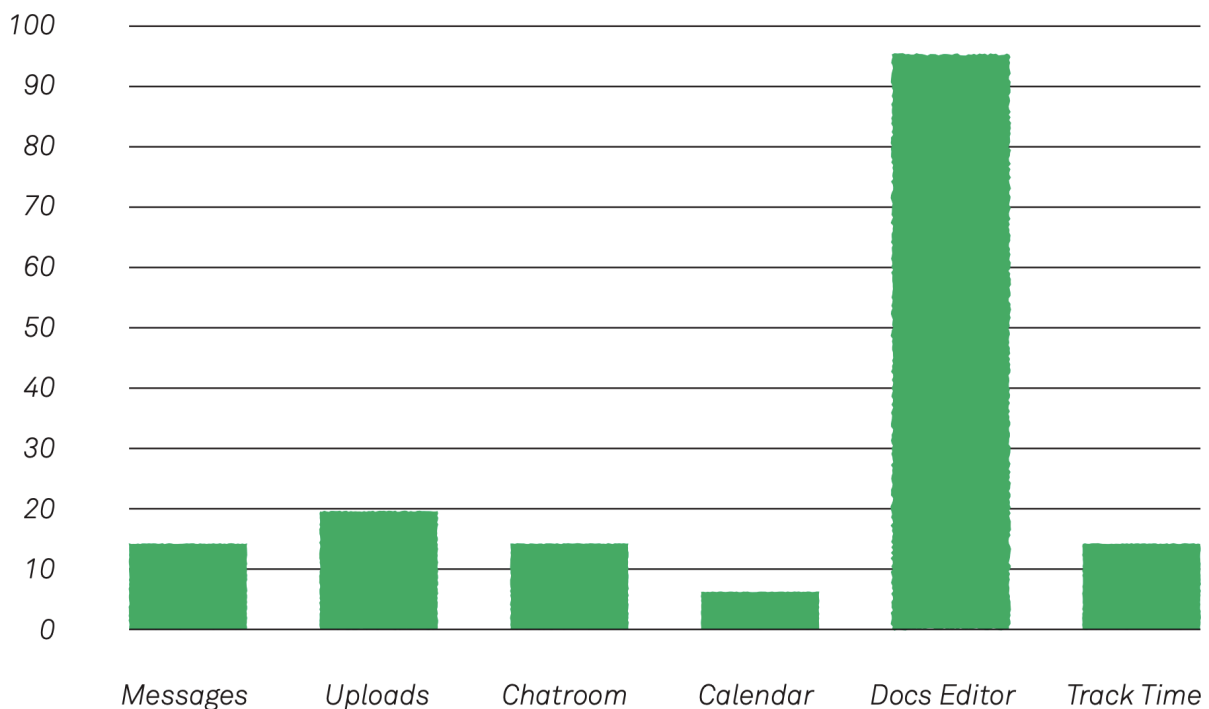
Как вы пришли к такому? Ведь вы создали продукт, в котором есть отличный мессенджер, редактирование файлов и документов. Эти ключевые функции выделены на вашем продающем сайте, четко объяснены скриншотами, есть в вашем интерактивном руководстве, и каждый новый пользователь любил их.

Но успех — плохой учитель. Вы были уверены, что не допустите ошибок и сможете создать гораздо больше полезных функций.

Поэтому вы добавили чат, но он не пошел у пользователей из-за неудачного релиза. Затем вы добавили календарь, и он пошел еще хуже. Никто не создал больше одного события, и о нем до сих пор ни слова на вашем продающем сайте. Вы добавили функцию учета времени, которая стала довольно популярной, но лишь у некоторых пользователей.

Это ваш продукт. Вы тот, кто может его исправить.

## Примечание о риске подрывных инноваций



*Распределение, указывающее на риск стать жертвой подрывной инновации*

Если вы наблюдаете такое распределение использования функций, это значит у вас отличный продукт для одного узкого случая использования. А всеми остальными функциями продукта никто не пользуется. Представьте Hangouts внутри Google+, в качестве примера, или дополнительный примитивный редактор документов (типа **BUI**) внутри Microsoft Word.

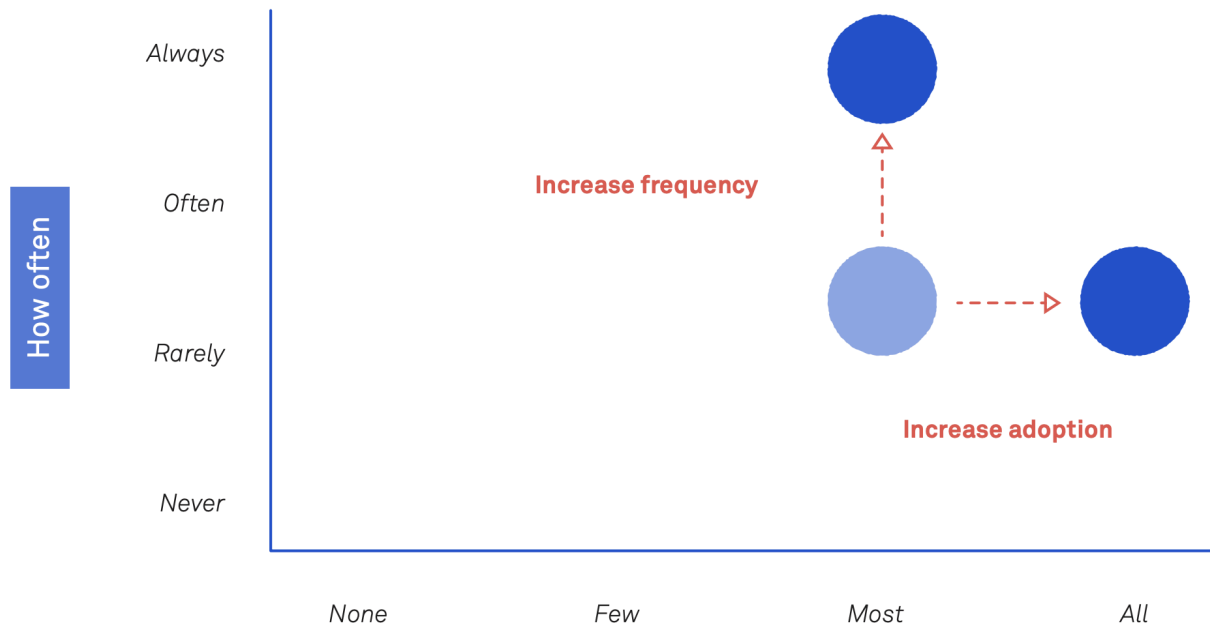
При таком распределении у вас большой риск стать жертвой [подрывной инновации](#). Кто-то может создать простой продукт с фокусом на этой единственной функции, которая будет лучше в чем-то одном (дешевле, быстрее, с совместной работой, удобнее, с поддержкой мобильных устройств и т. п.). И вы не сможете конкурировать, потому что поддерживаете все остальные ненужные функции.

## Что делать дальше с результатами аудита продукта?

Для каждой функции, которую мало используют, у вас есть выбор из четырех вариантов:

- ☐ Удалить: признать поражение и начать удаление из продукта.
- ☐ Увеличить уровень принятия: сделать так, чтобы больше людей пользовались ею.
- ☐ Увеличить частоту использования: сделать так, чтобы люди пользовались ею чаще.
- ☐ Улучшить: сделать ее значительно лучше, для тех, кто уже пользуется ею.

Это можно представить в виде диаграммы:



## Как улучшить функционал продукта

Кайдзен это философия непрерывного улучшения. Все онлайн компании в поиске продукта или рынка следуют Кайдзен в той или иной степени, даже если они не знают об этом.

Релиз нового кода не означает, что вы улучшили что-то. Аналогично, вы можете сделать неоспоримые улучшения, но при этом не получить отклика или одобрения от пользователей. Все зависит от того, какие улучшения вы делаете.

Два наиболее популярных способа улучшить продукт — добавить новые функции или улучшить имеющиеся. Сначала мы рассмотрим улучшения существующего функционала. Затем перейдем к созданию новых функций в следующей главе.

## Улучшение существующего функционала

Улучшить существующие функции можно следующим способами:

- ❑ Сделать их еще лучше (намеренное улучшение).
- ❑ Изменить их так, чтобы ими пользовались чаще (увеличение частоты использования).
- ❑ Изменить так, чтобы больше людей пользовались ими (увеличение принятия).

## 1. Намеренное улучшение

Это тот случай, когда вы знаете почему клиенты пользуются существующей функцией и почему она им нравится. Намеренное улучшение предполагает сделать функционал лучше именно для существующих пользователей.

Например, сделать функционал быстрее, удобнее или улучшить внешний вид.

**Используйте намеренное улучшение**, когда все ваши клиенты используют функцию и она им нравится. И вы видите возможность сделать ее значительно полезнее.

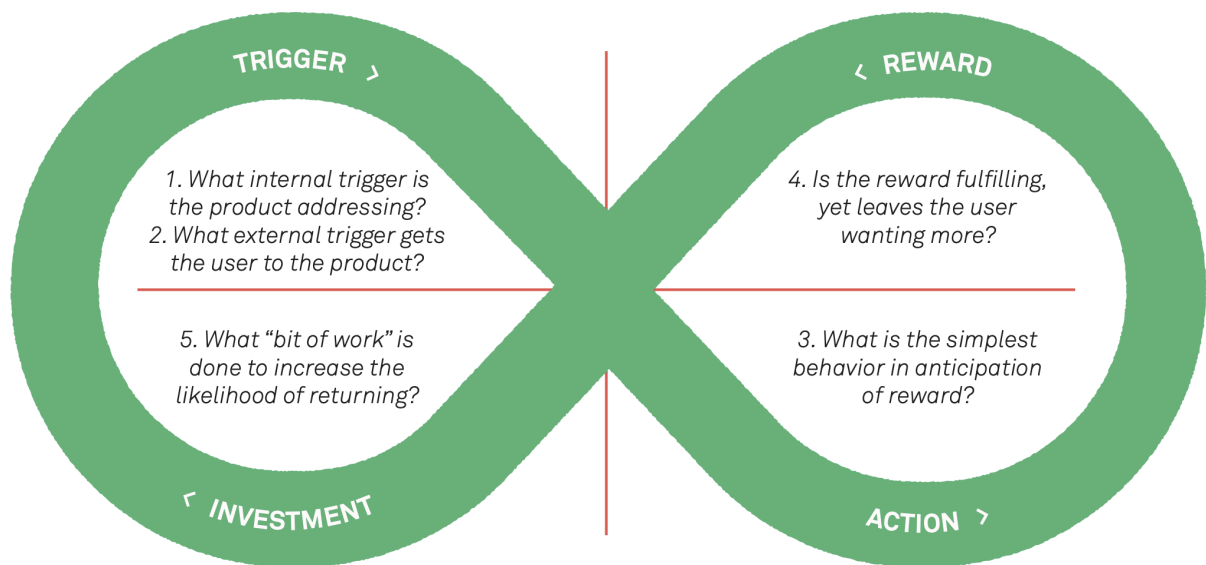
Стоит заметить, что намеренное улучшение часто используемой функции это одновременно большой риск и большой выигрыш. Представьте улучшение редактора в платформе ведения блогов. Сделали хорошо, и каждый пользователь выиграет от улучшения. Сделали плохо, и вы испортили работу всех пользователей. Большой риск, большая награда.

## 2. Увеличение частоты использования

Это улучшения с надеждой, что клиенты будут пользоваться функцией чаще. Добавление пунктов в ленте активности или больше настроек в поиске означает, что пользователи читают ленту чаще или используют поиск для большего круга задач. Такой тип улучшения может сделать функцию, которой пользовались раз в неделю, ежедневно используемой.

Функция подтверждения навыков в LinkedIn — хороший пример такого улучшения. Они добавили возможность перекрестного подтверждения навыков ваших друзей в один клик. Вы можете подтвердить навыки четырех друзей, и случайным образом получить подтверждение ваших навыков от других людей. В результате, это создает больше подтверждений, больше заходов в приложение и еще больше действий в один клик. Замкнутый круг.

Принцип, который они использовали, описал Нир Эяль, автор книги [На крючке](#). Он говорит, что привычка формируется из повторяющихся действий, состоящих из четырех ключевых элементов...



*Принцип формирования привычки*

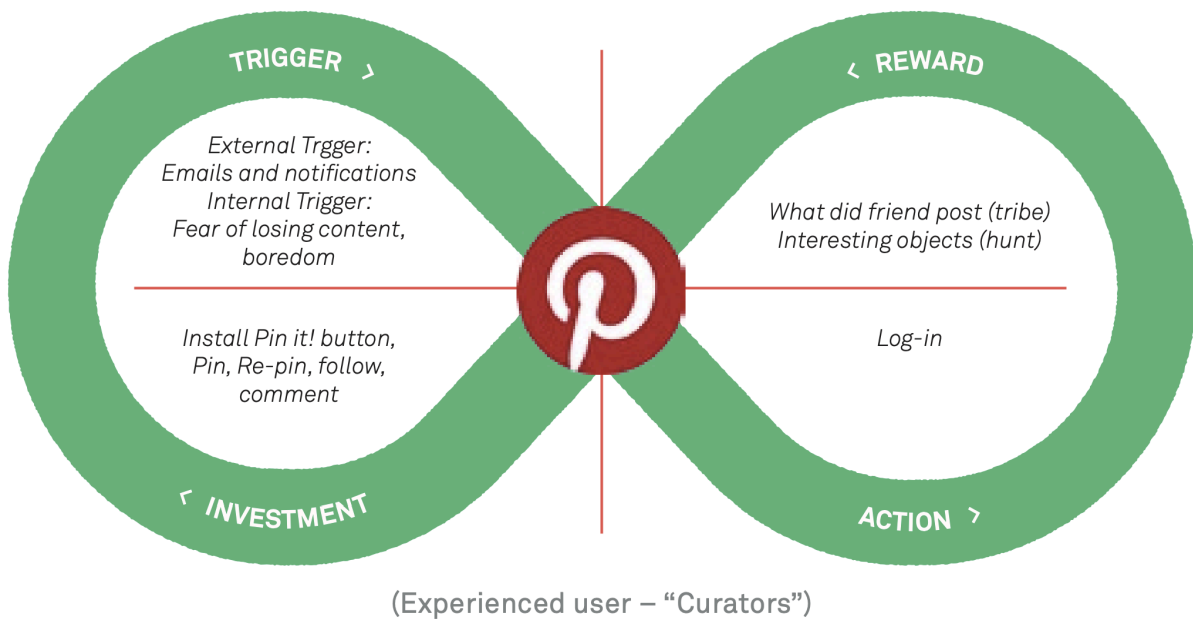
**Триггер:** причина, по которой пользователь заходит в приложение (получил письмо, о том что его навыки кто-то подтвердил).

**Действие:** которое пользователь совершает в ожидании награды (просматривает ленту, ищет и т.д.).

**Награда:** которую получает пользователь, от своих действий (видит красивую доску Pinterest).

**Инвестиция:** действие, которое приведет к большему количеству триггеров (подписка, лайк, установка связи с пользователями и т.д.).

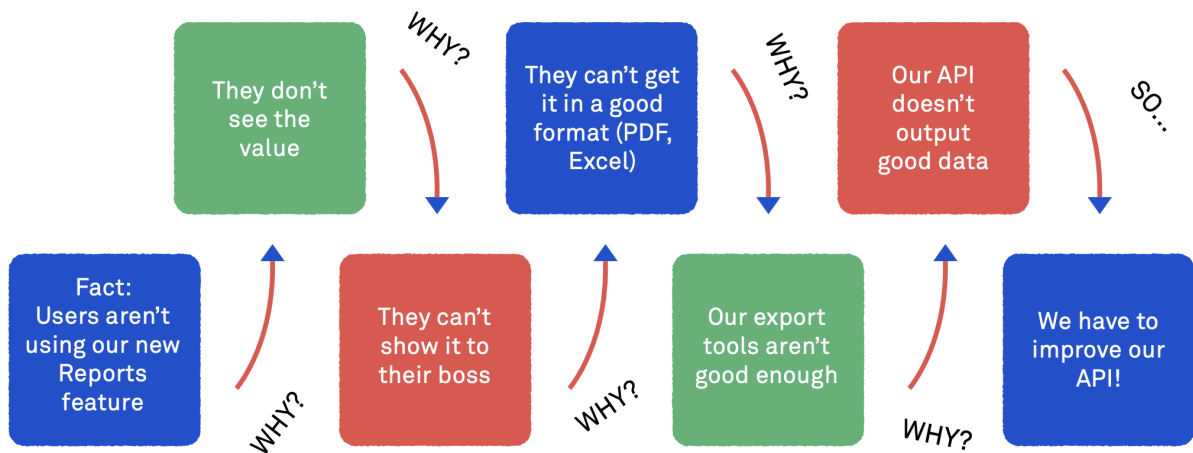
Вот как это выглядит на примере Pinterest



Формирование привычки в Pinterest

**Применяйте увеличение частоты использования:** для функционала, который **редко** используют **большинством** клиентов. Если вы верите, что частое использование будет полезно им. Имеет смысл оценить, какая польза для вашего бизнеса от увеличения частоты использования этой функции. Например, если Basecamp увеличит частоту, с которой пользователи создают проекты, они естественным образом увеличат прибыль, потому что их цена зависит от этого. При этом есть много возможностей сделать функцию привлекательной без пользы для бизнеса, а в некоторых случаях, даже ухудшить этим основное использование продукта. Например, показатели подтверждений навыков LinkedIn, вероятно, стали лучше. Но надо задать вопрос, не получено ли это ценой доверия к ним?

### 3. Улучшение принятия функционала

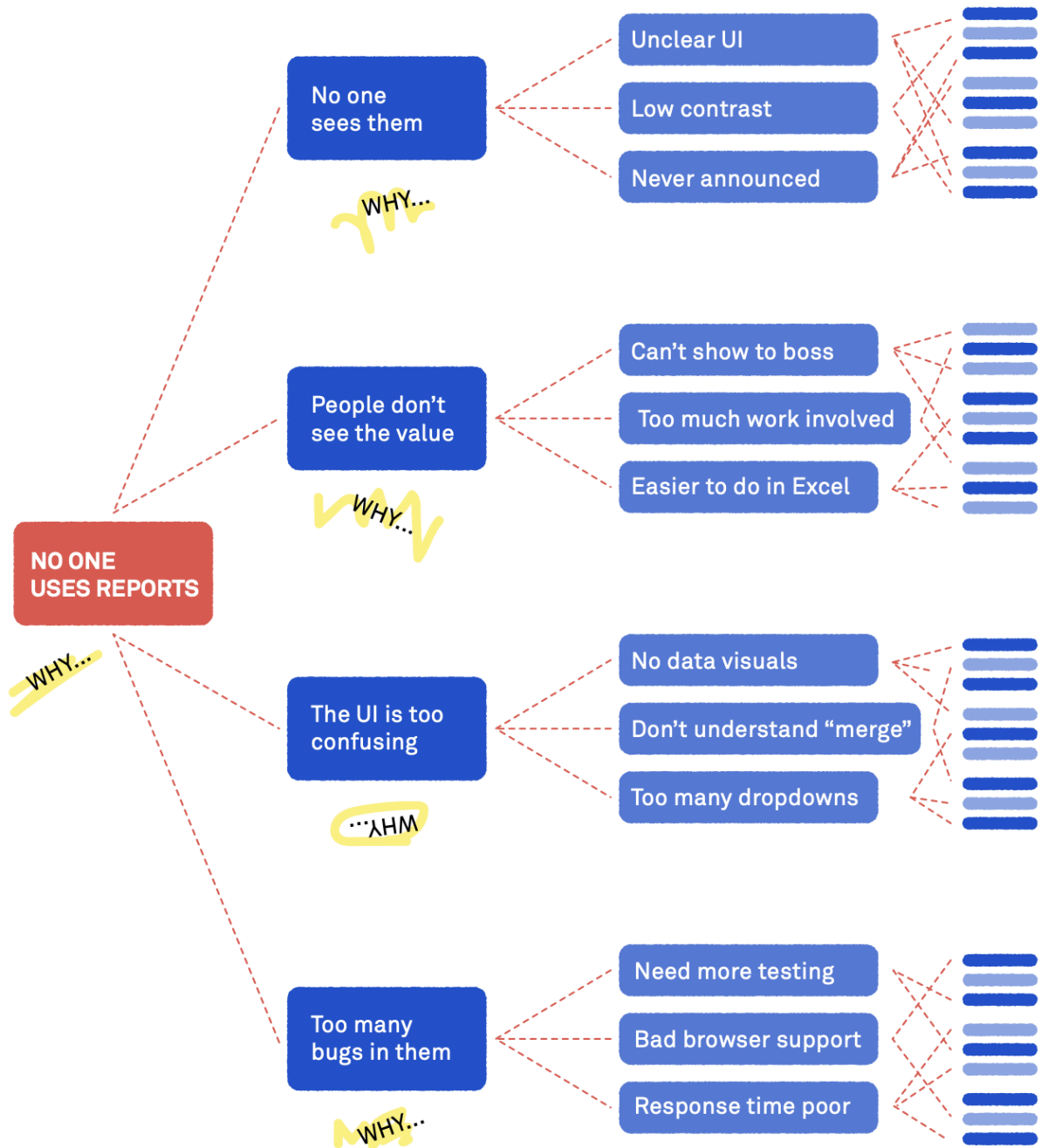


**Улучшение принятия** предназначено для тех, кто не пользуется функцией. Чтобы сделать ее полезной для большего количества пользователей, найдите и исправьте проблемы, которые мешают им пользоваться. Здесь хорошо работает техника [Пять почему](#).

Предположим, ваши клиенты не пользуются отчетами. Почему? Они не видят пользы в этих отчетах. Почему? Они не могут показать эти отчеты начальнику. Почему? Они не могут вывести в отчетах необходимые данные. Почему? Из-за плохого экспорта данных. Почему? Наш API не выдает нужные данные.

Если вы зададите вопрос достаточное количество раз, вы найдете корень проблемы.





Не ограничивайтесь разговором с одним клиентом. Скорей всего, причины более сложные чем описано выше. Вы обнаружите повторяющиеся причины. Затем, вы сможете исправить ключевые проблемы, которые не дают клиентам пользоваться всеми доступными функциями.

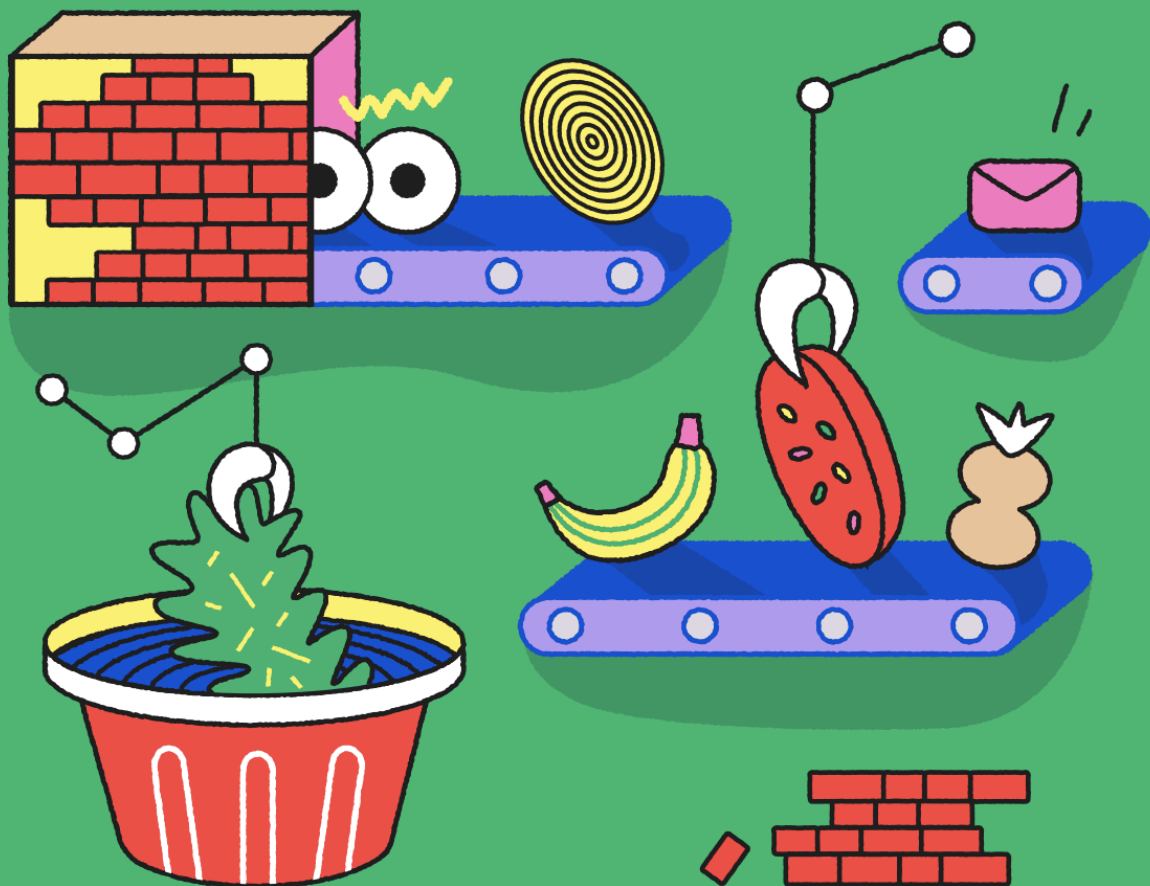
**Используйте улучшение принятия** если есть важная функция, которой не пользуются многие пользователи и вы видите способы, чтобы им было легче начать пользоваться.

Когда планируете улучшение принятия, также рассматривайте изменения вне кода. Иногда проблема не в том, как функция спроектирована или сделана, а в том, как она подается. Например, иногда пользователи должны знать почему и как использовать отчеты. В этом случае улучшение маркетинга продукта и коммуникаций с клиентами могут решить проблему без изменения продукта.

## Непрерывное улучшение

На ранних стадиях стартапы имеют преимущество перед существующими компаниями. Стартапы быстрее адаптируются к изменениям рынка. Потому что у них еще немного технических долгов, нет старого функционала, нет проблем с обратной совместимостью, нет дорогих клиентов, замедляющих их развитие. Но иногда эта скорость и гибкость может вызвать изменение направления как у безголовых куриц, вместо того, чтобы сфокусироваться на улучшении полезных функций их продукта.

Продакт менеджер в стартапах сталкивается с двумя основными сложностями. Первое, найти улучшения, полезные для бизнеса и клиентов. Второе, убедиться, что эти улучшения не потерялись в документации, а на самом деле реализовать их. Потому что для онлайн стартапов самое важное — выпускать изменения, иначе ты мертв.



## Глава 2. Когда говорить «нет» новому функционалу

Существует ограниченное количество улучшений, которые можно сделать в продукте или функционале. В какой-то момент вам придется решать, какие новые функции имеют смысл для продукта, и как вы будете подавать их пользователям.

Когда Facebook создал ленту новостей, это [вызвало бурю протеста](#) среди ранних и наиболее лояльных пользователей. Но был бы у Facebook такой рост, если бы он не проигнорировал протест и не показал бы ценность нового функционала?

В этой главе вы узнаете о создании дорожной карты продукта и формулу, показывающую, где лучше приложить усилия. И почему «нет» это самое важное слово в вашем словаре, которое вы должны говорить часто.

## Добавление новых функций

Новые функции расширяют возможности продукта, часто создают всплеск интереса, освещаются в некоторых СМИ. Эти фанфары привлекают новых клиентов и появляются новые варианты использования продукта. Обычно, новые функции — единственные улучшения, о которых когда-либо услышат не-клиенты.

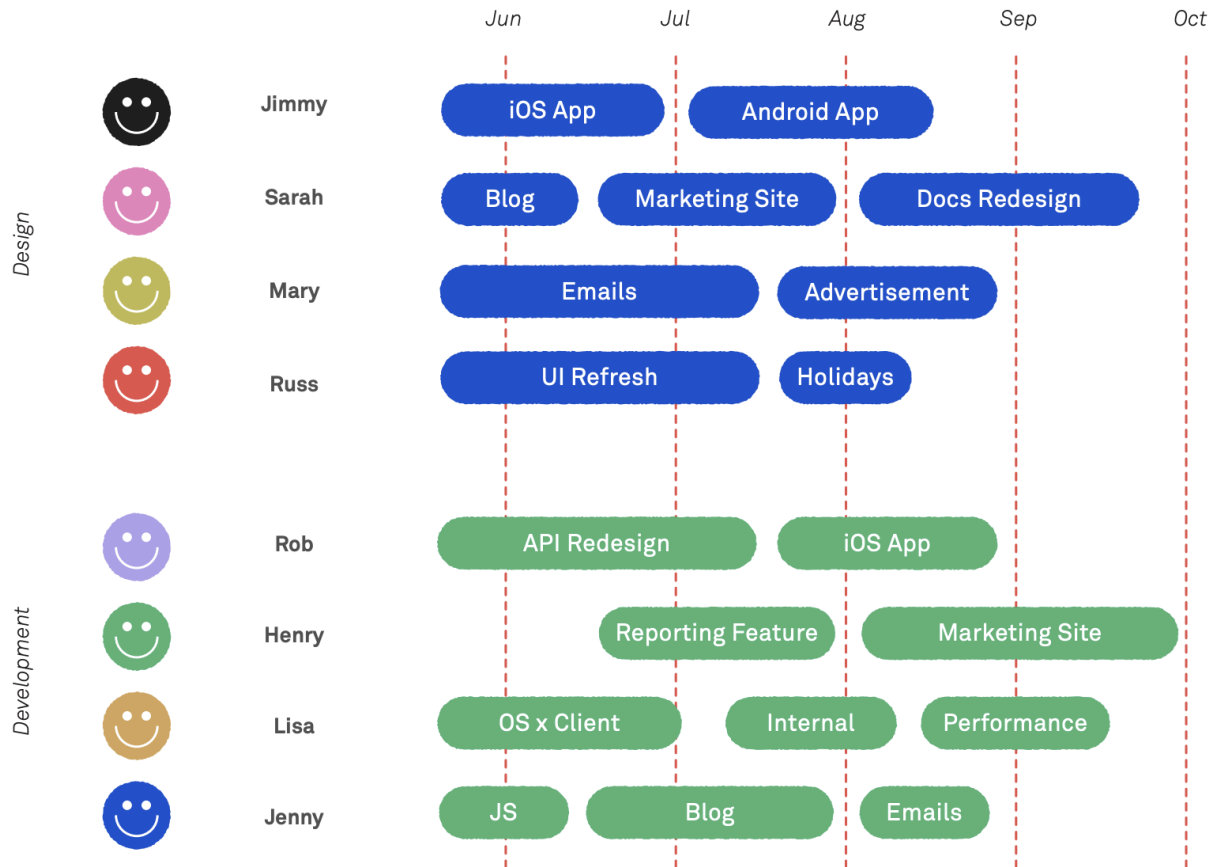
Но новые функции содержат в себе риски. Вы должны быть уверены, что они полезные. Потому что с ними, как с ребенком, вам придется поддерживать их, что бы ни случилось.

Спросите ваших клиентов «Хотели бы вы Календарь/Трекер времени/Диаграммы Ганта?» и они ответят «Да». Потому что это одностороннее предложение «что-то без ничего взамен». Клиенты не делали выбор между конкурирующими приоритетами. Это приводит к тому, что клиенты говорят «хочу» для функций, которые на самом деле им не нужны.

Спросите клиентов «Хотели бы вы, чтобы продукт был быстрее или, чтобы мы добавили функционал с метками?» и вы услышите другой ответ. Каждый ценит

скорость работы. Поэтому при планировании новых функций важно понимать компромиссы.

Где слабые места вашего продукта и где это имеет значение?



Пример дорожной карты продукта

Если вы фокусируетесь только на новых функциях, ваш продукт будет километры в ширину и несколько сантиметров в глубину. А если вы фокусируетесь только на исправлениях, у вас не будет инноваций, и ваш продукт устареет. Действительно, сложные решения.

Когда вам нужно улучшить продукт, полезно спросить, «Где наши слабые места и где это имеет значение?»

При улучшении продукта, вы фокусируетесь на наиболее важных частях продукта, работа которых не устраивает клиентов. Оба условия необходимы, иначе вы будете работать над частями, которые никому не нужны, или над теми, которые уже более чем хороши.

В своей известной статье [Создание инновации из отзывов клиентов](#), Энтони Ульвик предлагает алгоритм поиска возможностей, с практическими шагами создания дорожной карты, учета важности и удовлетворения потребностей.

Где находятся возможности?

	Imp.	Sat.	Opp.		Imp.	Sat.	Opp.
<i>Be aware of any problems with any of my projects</i>	9	7	11	<i>Keep our clients aware of what we're working on</i>	10	7	13
<i>Find what my team have been working on recently</i>	6	8	6	<i>Review and individual's contributions to a project</i>	5	8	5
<i>Understand how accurate our project estimates are</i>	4	1	7	<i>Record project decisions in a clear formal manner</i>	9	5	13
<i>Find contact details for a client quickly</i>	8	2	14				

$$\text{OPPORTUNITY} = \text{IMPORTANCE} + (\text{IMPORTANCE} - \text{SATISFACTION})$$

Алгоритм поиска возможностей Ульвика четко определяет недостатки в продукте. Это достигается тем, что пользователи ранжируют задачи, которые им нужно делать, по **важности** и насколько они довольны **существующим способом** делать эти задачи. В результате возможности определяются формулой:

$$\text{важность} + (\text{важность} - \text{существующий способ})$$

Эта формула часто выявляет возможности, которые не были замечены. Обычно самые большие возможности находятся в тех частях, о которых продакт

менеджер думает: «готово», «без ошибок», «достаточно хорошо» и т. д. Другими словами, небольшое улучшение в важной задаче почти всегда является большей возможностью чем большое улучшение во вспомогательной задаче.

Это тот случай, когда правило 80/20 может не сработать. Мысль, что 20% функций составляют 80% ценности продукта может и верная. Но это также значит, что вы предлагаете посредственное качество клиентам в наиболее важных для них задачах. Это наблюдение сделал Марк Цукерберг, когда рассказывал об истории Facebook...



Нельзя следовать принципу 80/20 во всем. Должны быть некоторые вещи, в которых вы лучшие. Вещи, в которых вы идете значительно дальше чем другие, поднимая планку на новый уровень и создавая самый лучший продукт, который только может быть.

Марк Цукерберг  
Основатель Facebook

Не существует единственного правильного способа приоритизировать функции в дорожной карте. Но зато есть много неправильных. Если есть возможности в существующих областях продукта, и ваша дорожная карта игнорирует их в пользу новых функций, ваш продукт вскоре может стать таким, который делает все подряд.

Разберитесь с проблемами в продукте, или это сделает кто-нибудь из конкурентов.

Почему вы не добавляете новые функции?

Продакт менеджеры и основатели в этой роли должны уметь говорить «нет». Не «может быть» или «позже». А только одно слово «нет».

Создание продукта это не значит добавление бесконечного количества полезных функций соответствующих тематике. А создание слаженного функционала с четко определенными параметрами.

Когда ваш продукт становится популярным, у вас появляется море хороших идей для нового функционала. Это идеи от клиентов, коллег и от вас.

Поскольку идеи хорошие, у вас будет много причин сказать им «да». Ниже 12 аргументов, которые обычно приводят, чтобы включить функции в продукт:

## 1. Но статистика выглядит хорошо

«Мы включили эту функцию для маленькой группы клиентов и степень принятия просто зашкаливает». Обычно такой подход страдает от выборочного анализа. Продукты — сложные системы. То, что выглядит как увеличение принятия функции на самом деле перемещение цифр с места на место.

Вам кажется, что ваша функция ярлыков это успех. Но заметили ли вы, что функция тэгов вдруг стала непопулярной? Даже если статистика показывает интерес и увеличение принятия функции, все равно нужно задать вопрос, соответствует ли этот функционал продукту? Добавьте Тетрис в ваш продукт и вы скорей всего увидите всплеск использования этой функции. Но стал ли продукт лучше?

## 2. Это займет всего пару минут

Объем работ никогда не должен быть причиной добавить функцию. Возможно, это причина добавить в дорожную карту как легкую победу, но это решение о дорожной карте, а не о продукте. Много плохих идей могут быть созданы быстро. Не дайте себя одурачить. Не существует маленьких изменений. Даже



небольшие изменения добавляют скрытую сложность, которую не учли в оценке «но это всего 5 минут».

### 3. Но мы потеряем клиента

Это разновидность шантажа. Ни один клиент не может быть важнее хорошего продукта. Путь к продукту сделанному на заказ уставлен табличками «сделано только для этого клиента». Он ведет к идеальному продукту, но лишь для одного клиента. И, несмотря на то, что они теперь зависят от вас, вы делаете то, что они скажут. Создание дополнительного функционала для одного клиента всегда идет в ущерб многим другим.

### 4. Но мы можем сделать это опциональным

Это ведет к смерти от обилия настроек. Делая функцию опциональной, безусловно, уменьшает сложность основного интерфейса, но эта сложность продолжает сквозить во всех остальных местах. Видимая цена за это — сложный интерфейс со множеством условных элементов и тоннами настроек. Скрытая цена — каждая опциональная функция ослабляет идею вашего продукта. Вы становитесь «трекер времени, который может рассылать инвойсы и сводить баланс платежей, но без отчетов».

### 5. Но сосед моего двоюродного брата сказал...

Это принятие решений на основе баек. Таких случаев полно в потребительских продуктах и в сервисных компаниях, которые никак не могут решить, что они все-таки делают. Ведь так легко на основе единичных случаев проигнорировать годы опыта, исследований, статистики.

Говоря «в компании моего брата они используют Google Analytics и расширенные сегменты» очень просто убедить использовать расширенные сегменты игнорируя вопросы:

- ☐ что делает ваш продукт?
- ☐ является ли компания вашего брата хорошим примером наших клиентов?
- ☐ действительно ли они используют или просто говорят?
- ☐ действительно ли расширенные сегменты - это хорошее решение для наших клиентов?

## 6. Но у нас сейчас ничего не запланировано

Дьявол подкидывает работу для незанятых команд, и получаются действительно ужасные функции. Если кто-то видит инженера, сидящего без дела, тут же дает ему создание новой функции, чтобы не бездельничал. Принимаются поспешные решения и делается дизайн на скорую руку во имя того, чтобы не терять время.

Это плохой путь «улучшения» продукта. Вместо того, чтобы добавлять технический долг, есть возможность уменьшить существующий долг. Кто работал на кухне знает: «есть время поспоняться, значит есть время прибраться». Свободное время лучше использовать для исправления ошибок, улучшения тестов, рефакторинга и др., а не разрушать идею продукта при помощи лозунга «чтобы команда была занята».

## 7. Но нам разрешили работать над чем мы захотим

Этот аргумент исходит из корпоративной культуры. Есть много крупных компаний, которые обещают инженерам создавать то, что они захотят и включать это в продукт. Обычно это обещание имеет одно из следующих последствий:

1. Это ложь, чтобы привлечь инженеров. Люди быстро замечают это и бросают попытки добавить свое в продукт. Вы не можете имитировать культуру компании.

2. Это правда. В результате получается продукт, полный незавершенных идей.

Есть разница между тем, чтобы вдохновлять инженеров создавать внутренние вещи (хороший вариант) и между тем, чтобы разрешать добавлять функции в продукт, минуя продакт менеджмент (плохой вариант).

## 8. Но 713 000 человек хотят эту функцию

Всегда найдется кто-то, кто будет использовать сырые числа, чтобы убедить вас. Любой продукт, которым уже пользуются, может сделать эмоциональное заявление наподобие такого: «Вы можете заполнить парк Долорес людьми, которые просят интеграцию с Excel». Вы действительно хотите сказать «нет» всем этим людям? Да, вы должны это сделать. В противном случае большинство ваших пользователей пострадают. Вопрос не в том, можем ли мы заполнить парк Долорес людьми, которым нужен этот функционал. А в том, является ли эта функция ценной в рамках видения нашего продукта для всех наших пользователей.

## 9. Но наши конкуренты уже сделали это

Это не означает, что это хорошая идея. Возможно они пробуют что-то. Это может быть ужасная идея. Это может быть что-то, что они планируют удалить из продукта. Ошибка предполагать, что ваши конкуренты значительно умнее вас. Одержимость функциями ваших конкурентов ведет к созданию «вчерашних технологий завтра».

## 10. Но если мы не сделаем, это сделает кто-то другой

Это не значит, что это должно быть в вашем продукте. Если кто-то делает это, станет ли клиентам не нужен ваш продукт? Уйдут ли они к конкурентам?

Говоря «кто-то другой» звучит хорошо, но ничего не значит. Мы все так иногда говорим. Часто эта логика используется для расширения продукта, потому что вы не хотите признать, что продукт должен иметь четкие границы. Вы боитесь прочертить эту линию.

Например, типичное свидание включает кино, ужин и такси домой. Если владелец кинотеатра постоянно беспокоится, что другие компании сделали и старается увеличить ценность своего предложения, он добавит ресторан в кинотеатр, и создаст компанию такси. В результате все три предложения будут слабыми. А рестораны начнут показывать кино...

## 11. Но начальник очень хочет включить эту функцию

Если начальник является также продакт менеджером и у него есть достаточно времени и знаний, чтобы принять целостное решение, тогда это хорошо.

Однако, если кто-то пытается заработать расположение руководства, занимаясь любительскими проектами, к которым склонен его менеджер, тогда это неизбежно ведет к проблемам.

## 12. Но это может быть «то что нам не хватало»

Это классический способ ссылаться на неизвестное. Создание продукта требует сложных решений о том, что включить в него. Вы можете строить домыслы о том, что любая не созданная функция могла бы принести успех продукту. Но это всего-лишь домыслы, ничего больше. Когда вы боитесь принять трудные решения, вы скатываетесь к подобным домыслам, и, в результате, добавляете в

продукт все подряд. В конечном итоге у вас получается набор функций, а не продукт.

## Почему «нет» это важно

Дело в том, что никто не держит плохие идеи в дорожной карте. Определить и избежать плохих идеи несложно. Но настоящие решения о продукте — непростые. Они требуют посмотреть на предложение и сказать: «Это действительно хорошая идея. И я понимаю, почему наши клиенты хотят ее. Но мы не будем делать ее. Вместо этого, вот что мы сделаем.»

## Не бывает маленьких изменений

Все еще не верите, что маленьких изменений не бывает? Давайте рассмотрим следующий пример.

«Мы хотим ограничить длину текста в комментарии до 140 символов, потому что нам может потребоваться использовать SMS для отправки на определенном этапе. Это маленькое изменение, верно?»

Не верно!

Не бывает маленьких изменений, если вы настроены делать качественный продукт. Конечно, программист может сделать это за три минуты. В конце концов, это всего лишь одно условие в коде.

Но продакт менеджмент никогда не бывает таким простым. Прежде чем приступить к «маленьким изменениям», нужно задать несколько вопросов. Возвращаясь к длине комментария в нашем примере, давайте начнем с нескольких простых вопросов.

Что будет, если длина комментария будет больше 140 символов? Должны ли мы просто обрезать строку, или показать пользователю сообщение об ошибке? Если показываем сообщение, где оно появится? Что в нем говорить? Кто напишет текст сообщения? Как мы объясним пользователю, почему мы ограничили длину комментария до 140 символов? Как будет выглядеть сообщение? Определен ли у нас стиль для подобных сообщений? Если нет, кто подготовит дизайн?

Но подождите, есть кое-что еще...

Если даже у нас есть ответы на все эти вопросы, осталось кое-что еще. Делать это только на стороне сервера не самый лучший способ обработки ошибок. Нужно сделать это также и на клиентской стороне. Но если мы будем делать проверку ошибки на клиентской стороне, возникают новые вопросы...

Кто напишет изменения в JavaScript? Будет ли JavaScript код показывать ту же самую ошибку, что и серверный код? Как оно ведет себя с отключенным JavaScript? Что сделать, чтобы в будущем при любых изменениях правила 140 символов не забывать менять и на серверной, и на клиентской стороне?

Но мы еще не закончили. Посмотрите на это с точки зрения пользователя. Они недоумевают, почему есть ограничение в 140 из-за странной причины, которую они не понимают. А теперь мы еще требуем от них угадывать длину сообщения? Должен быть способ лучше чем угадывать. Давайте добавим счетчик символов. О, это порождает еще несколько вопросов...

Почти завершили...

Кто сделает счетчик символов? Если будем использовать готовый, найденный в интернете, кто протестирует во всех браузерах, которые мы поддерживаем?

Также, где будут отображаться цифры счетчика? Как будет выглядеть счетчик? Конечно, нужно изменить стиль цифр, когда пользователь приближается к нулю, и определенно, цифры должны выглядеть по другому, когда

пользователь написал больше 140 символов. А может нужно не давать вводить больше? Если так, что будет, если они вставят текст из буфера обмена? Должны ли мы позволить редактировать, или предупредить их?

Мы сделали счетчик, отображение ошибок, добавили проверку на стороне сервера, и протестировали во всех поддерживаемых браузерах. Теперь осталось написать автоматические тесты и выпустить новый функционал. Предполагая, что механизм выпуска новой версии у вас отлажен, это сделать несложно.

Все это игнорирует факт, что пользователи будут недоумевать, почему, кто-то написал большой комментарий некоторое время назад, а они теперь могут писать лишь 140 символов. Очевидно, что мы должны предупредить техподдержку об этом изменении, обновить документацию, API, приложения для iPhone и Android. Также, нужно решить, что делать со старыми комментариями, урезать их или оставить как есть?

Не говоря уже о том, что делать со всеми этими эмодзи, которые так популярны сегодня... удачи в отправке их через SMS. Возможно, нам нужно не разрешать вводить недопустимые символы, а это значит новые сообщения об ошибках, дополнительный код... список задач продолжается.

Когда вы прошли все это, ваша новая функция будет в продукте, и это лишь для счетчика символов. Теперь попробуйте что-нибудь посложнее. Не существует маленьких изменений, если вы делаете вещи как следует. Вот почему, как продакт менеджер, вы должны хорошо понимать, что стоит добавить функцию, прежде чем вы кивнете головой и добавите функцию в дорожную карту.

## Учет всей картины

То, что кажется двухминутной работой, часто превращается в двухчасовую работу, если не учитывается вся картина. Функционал, который выглядит как

«полезный» всего за пару минут можно смело оценивать как больше двух часов.

**Вывод:** Объем работы скрывается в минутах, а не в месяцах. Следите за минутами, и будет порядок с месяцами. Дать согласие на добавление функции обманчиво легко. А вот программирование этих функций уже вряд ли будет легким делом. Поддержка будет кошмаром. Когда вы стремитесь к качеству не бывает маленьких изменений.

## Ничего не появляется бесплатно

В разработке программ вы часто можете услышать, что получаете что-то бесплатно. Например: «фреймворк дает это бесплатно» или «если мы перейдем на провайдера X то получим Y бесплатно». Ваши разработчики могут сказать: «мы итак это делаем, поэтому получим этот дополнительный кусочек бесплатно».

Ничего не появляется бесплатно в разработке продукта.

Это не бесплатно, потому что вы сидите и разговариваете об этом, дискутируете стоит ли делать это. Это затраты времени. Это не бесплатно проверять, что «бесплатный» кусок сделан правильно, и что он работает во всех сценариях. Тестирование никогда не было бесплатным.

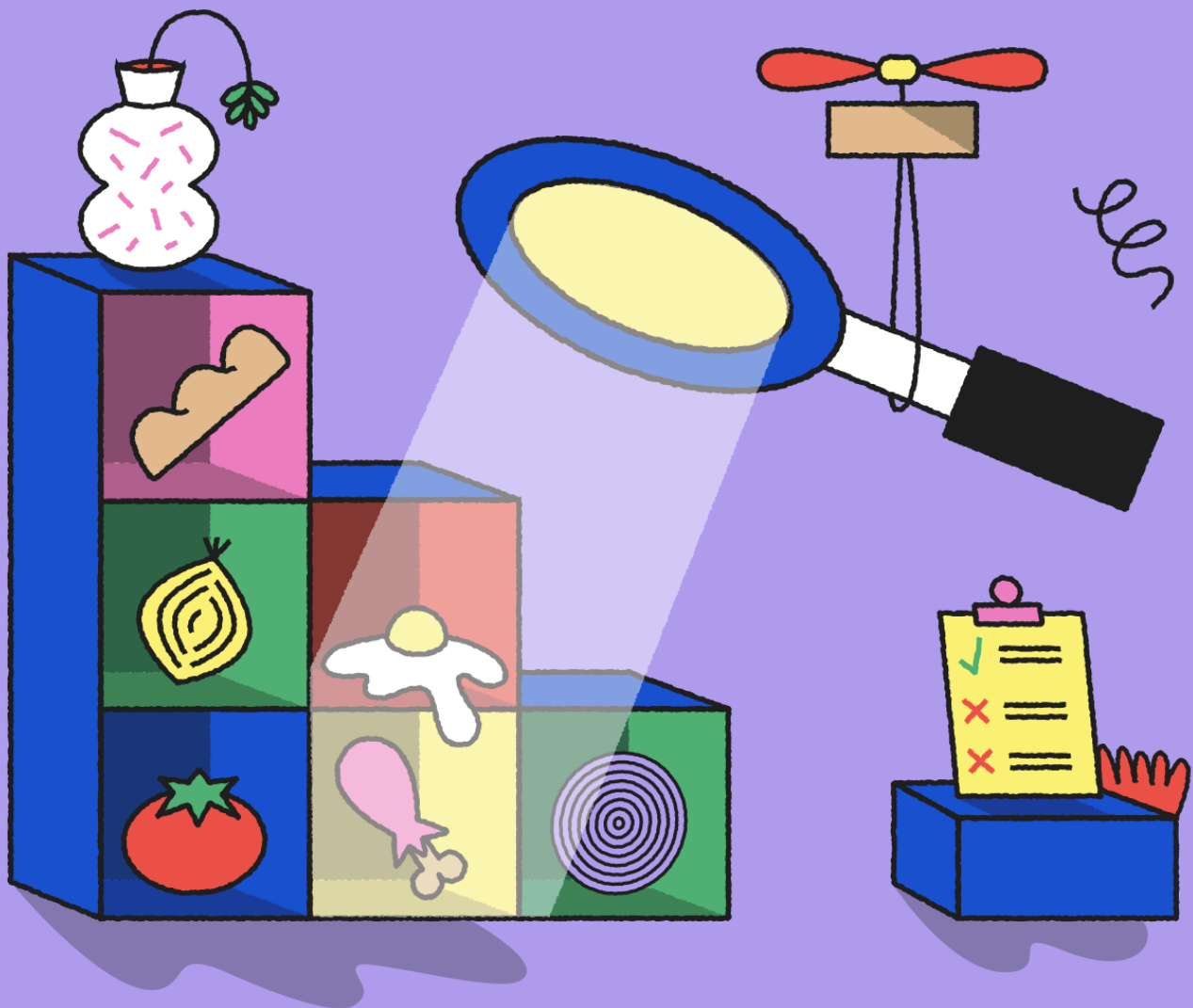
Это не бесплатно обсуждать добавление функции. Если вы добавите функцию, вам нужно сообщить вашей команде, поддержке клиентов, и самим клиентам. Вы должны сообщить клиентам, иначе это действительно бессмысленно. Коммуникации не бесплатные.

Это не бесплатно содержать функцию. Возникают вопросы от клиентов в службу поддержки. Вам необходимо обновить документацию, видео, все должно быть обновлено.



Даже удалить функционал не бесплатно. Поддержка никогда не была бесплатной.

Функционал можно сравнить с домашними животными. Полно объявлений о бесплатных животных. Но вы не берете их всех домой по одной простой причине. Есть большая разница между ценой покупки и стоимостью содержания.



## Глава 3. Какие функции добавлять в продукт

Когда дело касается стратегии продукта в Интерком, нас выручает техника [Работа-которую-надо-сделать](#) (Jobs-to-be-done). Эта техника помогает решить, что добавлять в продукт. При помощи этого подхода выясняются причины использования продукта клиентами, вместо традиционной техники сегментирования клиентов по демографическому признаку.

Люди не покупают продукт, потому что они попадают в какую-то группу, например, женщины, после 30 лет, живущие в частном доме, работающие неполный день. Они покупают, чтобы решить конкретную проблему. Если вы понимаете «работу» для, которой клиенты «нанимают» ваш продукт, в этом случае вы сможете сфокусироваться на том, чтобы они получили желаемый результат. Функции, которые вы добавляете должны помогать клиентам сделать их работу.

## Как сделать вещи, которые нужны людям

Проблемы, с которыми сталкиваются люди в жизни, редко меняются из поколения в поколение. Но продукты, которые они используют для решения проблем — меняются постоянно.

Когда вы создаете новый продукт, вы должны верить, что сможете сделать лучшее решение, которое люди захотят использовать, потому что оно даст лучшие результаты. Четкое понимание какой результат хотят клиенты и как они достигают его сейчас является основой, чтобы преуспеть в продакт менеджменте.

Может ваши клиенты хотят развлечений, или проводить больше времени с друзьями, или знать, над чем работают их коллеги, а может им нужно увеличить рост их бизнеса. Если желаемый результат реальный, они уже достигают его тем или иным способом. Ваша задача — улучшить этот способ.

**Примечание:** Если вы не можете узнать, какой способ они используют сейчас, есть шанс, что это выдуманный результат («Было бы круто иметь все мои файлы из социальных СМИ в одном месте») или просто желание («Конечно я хочу похудеть»).

Фокус на результате вместо категории, индустрии или типе продукта, позволяет узнать ваших настоящих конкурентов. Компания, которая сосредоточена на отрасли, в которой она находится, а не на результатах, которые она приносит, теряет связь с реальностью, и вскоре теряет клиентов.

Например, газеты верили, что они в газетной отрасли. Как результат, они не могли понять, почему скучающие люди по дороге на работу перестали покупать их. Они смотрели по сторонам, пытаясь понять, кто из их конкурентов украл у них клиентов. Они экспериментировали с новыми форматами, низкими ценами, броскими заголовками, но не смогли остановить упадок. Если бы они сфокусировались на результате (скучающие люди хотят развлечься в пути в короткие промежутки времени при помощи мини-статей). Тогда их конкуренты (Twitter, Facebook, новостные приложения) не были бы так опасны для них.

Давайте рассмотрим несколько областей, которые как в случае скучающих людей в транспорте, остаются актуальными, благодаря сотням технологических улучшений.

Люди, в частности молодые люди и студенты, хотели передавать записи и сообщения, без страха, что другие люди увидят их...

*Люди все еще хотят сделать это, и они пользуются Snapchat.*

Люди хранили фотографии в безопасном месте, таком как коробка из под обуви под кроватью.

*Люди все еще хотят делать это и пользуются Dropbox.*

Люди хотели ставить свои любимые фотографии на видном месте, например, над камином, чтобы все могли видеть их...

*Люди все еще хотят делать это и пользуются Facebook.*

Люди хотели собирать вырезки из газет и журналов с идеями о реновации дома или другими проектами...

*Люди все еще хотят делать это и сегодня они пользуются Pinterest.*

Люди хотели делиться отзывами и советами для других путешественников в гостевых книгах...

*Люди все еще делают это и сегодня они используют Foursquare.*

## Делать работу лучше

Существуют сотни примеров как те, что показаны выше, и в них есть закономерность. Чтобы люди захотели что-то, необходимо понимать истинную причину потребности людей или бизнеса и, затем, использовать технологию для того чтобы:

- ☐ упростить шаги достижения результата,
- ☐ сделать доступным для большего количества людей,
- ☐ сделать доступным в других ситуациях.

Первый подход, упрощение шагов достижения результата, часто используют стартапы. Взять потребности, где существующие решения устарели или слишком сложные, и найти более простые шаги для достижения того же результата. Заказ такси в городе предполагал звонки на несколько номеров, пока не находилась компания со свободной машиной, затем длинный диалог о том, где вас забрать, куда отвезти и во сколько вам нужно быть на месте.

Сегодня вы нажимаете кнопку и машина появляется.

Второй подход обычно предполагает снижение затрат времени или более низкую цену, или уменьшение барьера использования продукта, чтобы больше

людей могли пользоваться им, и таким образом, расширить рынок.

Четырнадцать лет назад, если кто-то хотел вести блог онлайн, он должен был арендовать Linux сервер, загрузить на него архив с программой ведения блога, выполнить серию странных команд для распаковки и конфигурации.

Сегодня это несколько кликов на Medium.

Третий подход убирает ситуационное ограничение. Прием платежей раньше требовал массивные компьютеры с рулонами термобумаги, отправкой документов в банк по факсу, ISDN линиями и пакетными транзакциями, выполняемыми по ночам.

Сегодня карточки можно принимать даже при помощи смартфона.

У Джефа Безоса есть знаменитое высказывание «сосредоточьтесь на вещах, которые не меняются». Проблемы, с которыми сталкиваются люди и компании меняются не очень часто. Способы, которыми их решают, меняются почти каждый год. Таким образом, создание продукта, который люди будут использовать, должно начинаться с того «что хотят люди», а не заманчивого «то что мы можем».

**Запомните:** Легче сделать вещи, которые хотят люди, чем заставить людей хотеть вещи.

## Лакмусовый тест для новых функций

Ниже список вопросов «да/нет», на которые вы можете быстро ответить, прежде чем добавить новую функцию в дорожную карту.

В предыдущей главе мы рассказали, как стратегия продукта помогает сказать «нет». Но список причин, по которым мы отказываем в добавлении функции, не особо помогает понять, стоит ли добавить функцию. Поэтому ниже список вопросов, на которые должны быть ответы «да».

## 1. Соответствует ли функция видению продукта?

Во что вы верите? Что вы знаете о проблеме, о чем не знают другие? Как, вы считаете, должна быть решена проблема? Любой может найти статистику, запустить фокус группы, прочитать отчеты Gartner, выйти из офиса. Но у вас есть свое видение.

Решения, основанные только на видении, иногда кажутся иррациональными, потому что это сложные решения. Люди понимают, что их трекер ошибок не умеет отслеживать время доступности сервера. Никто не ожидает этого от трекера ошибок. Но должен ли инструмент для ведения проектов иметь отчеты? Уже не так очевидно.

Чем больше нюансов в решениях, тем больше сопротивление. Коллеги, клиенты, даже другие продакт менеджеры и основатели компании будут сопротивляться. Вот несколько примеров:

- ❑ Apple отказалась создавать нетбуки в то время, когда нетбуки были самыми популярными. Каждый аналитик просил об этом.
- ❑ Basecamp отказались добавлять диаграммы Ганта в свой продукт. За это их прозвали слепыми идеологами.
- ❑ Разработчик Гаррет Димон отказывается добавлять статус «отложить» в его трекер ошибок — Sifter. Потому что он не считает, что это правильный способ работать с ошибками.

Что еще хуже, вы никогда не узнаете были ли вы правы. Здесь не существует верных или неверных решений. Это искусство, а не наука. Это лишь вы и ваше видение.

## 2. Будет ли это актуально через 5 лет?

Это трудный вопрос, будет ли эта функция полезной через 5 лет? Вы будете чувствовать себя старым брюзгой. Но приложение, которое было популярно в 2013 исчезло в 2014. Все эти эффекты скольжения, растворения и складывания, которые так все любили год назад, выглядят как дурной вкус сегодня. Если вы хотите потратить лучшие годы своей жизни на продукт, держитесь подальше от модного, сосредоточьтесь на смысле.

## 3. Для всех ли это полезно?

Не забывайте о ловушке «слышал часто/недавно». Вы не сомневаетесь, что вещи, которые вы слышите часто или недавно, должны быть добавлены в дорожную карту. Это естественная реакция, вызванная вашим сопереживанием клиентам. Это не вежливо, говорить «нет» людям, и слышать эти просьбы снова и снова. Поэтому «слышал часто/недавно» бывает оправданием. «Конечно мы добавим это, я слышал это сегодня уже дважды», говорит директор с 4800 активными пользователями, для удовольствия его 0.0625% пользовательской базы.

Опасность реакции на «слышал часто/недавно» в том, что оно дает иллюзию анализа, ощущение, что вы сделали домашнюю работу и это результат рационального вывода, к которому вы пришли. В действительности вы приняли ленивое решение для удовлетворения мимолетного желания небольшой группы пользователей, не затратив и секунды на исследование, сколько на самом деле пользователей хотят или нуждаются в этой функции.

## 4. Это улучшение, дополнение или инновация существующего процесса?

Создание полностью нового процесса в вашем продукте должно быть редким случаем. Большинство времени, должно быть инвестировано в улучшение,



дополнение или инновацию существующих процессов. И в каждом случае вы должны знать, что из перечисленного вы делаете.

Если вы улучшаете существующее решение, вашими метриками могут быть: довольство клиентов, уменьшение [времени на использование инструмента](#) или количества обращений в поддержку.

Если вы дополняете существующее решение, метриками могут быть увеличение частоты использования, потому что теперь оно работает для новых случаев и для новых условий.

Инновация решения — самое сложное. Вы целитесь в мифический «совершенно новый способ», который содержит много рисков, но дает большой выигрыш. В этом случае можно измерять количество новых клиентов, хотя они и будут получены неявно, как результат новостей или маркетинговой кампании.

Обновления это всегда увлекательно, но вы можете впустую потратить время. Хороший способ определить стоит ли делать, просто спросить: «Будет ли больше людей использовать это? Будут ли люди чаще этим пользоваться? Если ни то ни другие, будет ли это, действительно, лучше для тех, кто уже использует это?»

## 5. Влияет ли это на рост бизнеса?

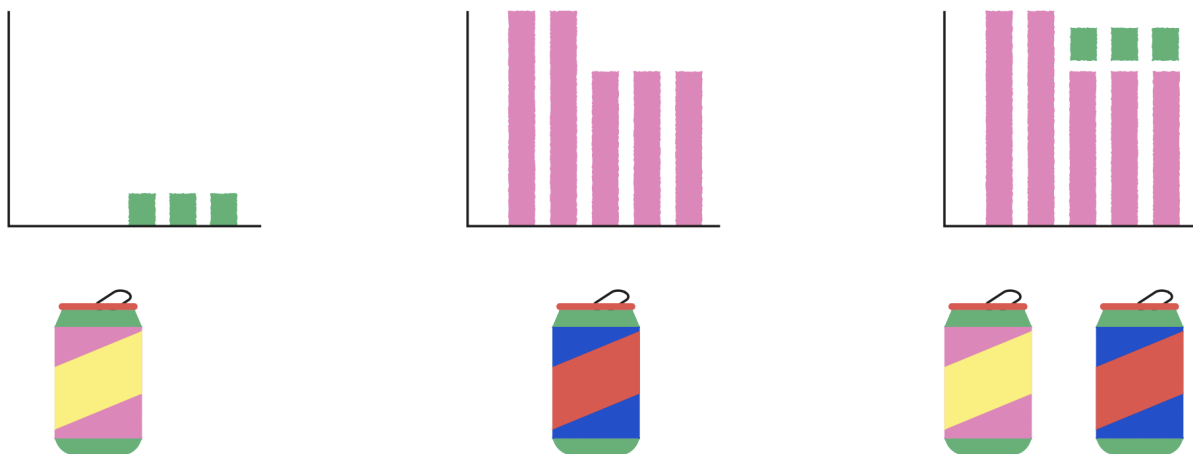
Можете ли вы увидеть какое влияние окажет функция на прибыль? Например, в приложении для управления проектами, шаблоны проектов могут стать таким случаем. Шаблоны могут помочь использовать продукт в необычных для клиентов случаях. Это увеличит количество проектов, которые создают клиенты, и, в свою очередь увеличит количество переходов на дорогие тарифы.

Обратите внимание, уменьшение уходов клиентов, также влияет на рост бизнеса. Неважно откуда пришли доллары. Часто функции добавляются, чтобы

удержать клиентов. Идея всех этих случаев в том, чтобы понимать, как любая работа влияет на рост бизнеса, в конце концов, все работают для роста.

6. Создаст ли это новый осмысленный вид вовлеченности клиентов?

Большинство метрик игнорируют систему и фокусируются на отдельной функции. Добавьте новую кнопку в продукт и люди будут нажимать ее. Получили достаточно кликов и вы можете назвать это увеличением вовлеченности. Но в этом нет особого смысла. Противоположная метрика может быть «уменьшилось ли использование чего-то другого?». Если вы добавили метрику для одной области продукта, необходимо анализировать другие области, которые могут быть затронуты.



*Следите за побочными эффектами*

Метафора из реального мира может быть влияние Диетической колы с лаймом на продажи Обычной диетической колы

На первом графике видно, как запуск новой колы повлиял на продажи. Если вы продакт менеджер Диетической колы с лаймом, вы можете назвать это успехом. Но если вы директор компании, вы поймете, что усложнили процесс производства с закупкой всех этих лаймов, соковыжималок и рекламой. И при этом общая прибыль не увеличилась. Не совсем успех. Не важно какие метрики вам показывал продакт менеджер.

## 7. Если это хорошо пойдет, сможем ли поддерживать и содержать это?

Одно из заблуждений о «быстрых победах» и «легких изменениях» это то, что учитываются только усилия на создание. Например, кто-то удивил вас JavaScript закладками для вашего приложения, или агентство создало 38 приложений для Windows Phone в рекордные сроки и по низкой цене. Поскольку усилия были сравнительно небольшими, вы решаете выпустить это. Успех!

Затем появляются запросы в техподдержку. Оказывается никто из команды не имеет достаточно опыта с новым для них фреймворком, который использовали для приложения. И вы застряли с поломанным продуктом, и несколько сотен клиентов жалуются в поддержку. Конечно, это никогда не бывает таким очевидным. Хорошие инженеры редко скажут, что не знают что-то. Они разберутся с этим за пару вечеров и дадут оценку, когда исправят ошибку. Но это затраты времени, о которых вы не думали когда выпускали продукт.

Другая область где можно обжечься — делать предложения которые вы не можете потянуть. Если у вас есть система CMS, и вы предлагаете бесплатный дизайн главной страницы, вы получите больше регистраций пользователей. В этом есть смысл на ранней стадии развития компании, когда вам действительно нужно соблазнить клиентов попробовать новый продукт. Но это не работает в долгосрочном периоде. Очевидно, что делать вещи, которые не масштабируются, это замечательная стратегия, до тех пор пока не нужно масштабироваться.

## 8. Можем ли мы спроектировать это так, чтобы выгода была больше чем усилия?

Как [писал мой коллега Пол Адамс](#), для любой функции, чтобы ею пользовались, получаемая ценность, должна быть больше чем усилия на

использование. Пользователи понимали преимущества, которые может дать Google Plus. Но усилия на регулярное перетаскивание множества контактов в разные квадраты просто не стоили этого. Я чувствую, что [приложения по разделению чека](#) попадают в эту же категорию. Разделение чека может быть проблемой, но любое решение, которое было до сих пор, стоит гораздо больше, и речь не о цене приложения. Речь о времени и усилиях.

Проектирование продукта — это анализ затрат/выгод. Насколько полезно что-то по сравнению с затратами на использование функции. Вот как это можно представить...



Важно знать, над каким квадрантом вы работаете. И будет мудрым держаться подальше от левого верхнего квадранта, оставив его лишь для нескольких базовых функций.

9. Можем ли мы сделать это качественно?

Каждый продукт имеет слабые стороны, часто в тех областях где продакт менеджер робко пробует и терпит поражение. Но эти поражения редко приводит к удалению функции. В результате, остаются «грязные следы» и запутанная идея продукта.

Проблема возникает, когда команды затрагивают области, в которых они слабо разбираются. Часто это случается, когда команды выходят за пределы «[сделано для себя](#)», когда они еще были сами себе клиентом. В этот момент их процесс проектирования ломается, и им нужен новый.

Например:

- ❑ Команды, которые не отслеживают время, добавляют функцию отслеживания времени.
- ❑ Люди, которые не ведут календарь проектируют функцию управления календарем.
- ❑ Люди, которые не пользуются трекером ошибок, создают систему ведения ошибок.

Обратите внимание, ничего из перечисленного не является неправильным. Неверный лишь процесс проектирования. Его нужно изменить от «для меня сойдет» в сторону фокуса на проблеме. Чтобы сделать хороший функционал, необходимо глубоко понимать задачу. Как в старой поговорке: если не можете сделать это хорошо, не стоит это делать.

#### 10. Можем ли мы определить минимальный объем работы?

Жизненно важно — начать с [минимальной реализации](#) функции. Ранние релизы дают обратную связь, необходимую для развития идеи. Признак того, что объем работ выбран неверно, отсутствие спецификации. Например, «чтобы работало для крупных компаний». Другой признак, если работа описана с

точки зрения функции, а не результата, который необходимо достичь. Например, «отчеты», вместо «чтобы менеджеры могли видеть продуктивность их команды».

Легко согласиться с различными клише на совещаниях. Кто-то скажет «это должно быть удобнее для крупных команд». Каждый кивает, и запись об этом появляется на белой доске. Только в специфике можно определить объем работ. Например, «нужно дать возможность группировать коллег по отделам» вместо «нам нужна поддержка крупных компаний».

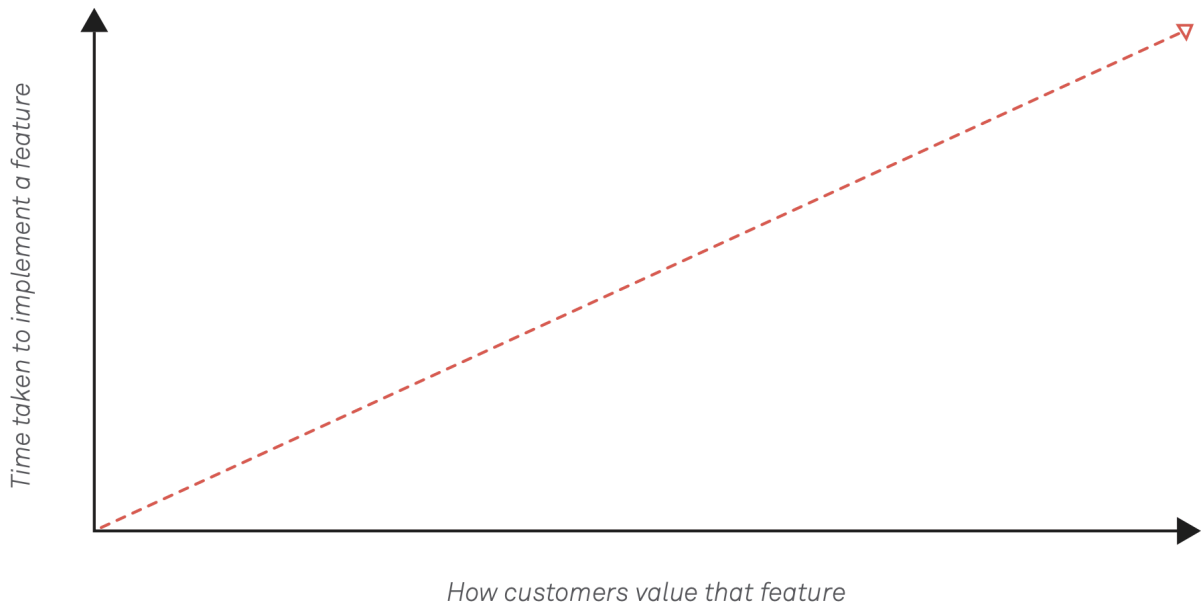
Функции с плохо определенным объемом работ не выполняют ожиданий пользователей, выпускаются поздно, если вообще выпускаются, и не добавляют в продукт ничего, кроме запутанности.

## Скользкая дорожка игнорирования правил

Соблазнительно иногда нарушать правила из списка выше. Можно предположить, что редкое нарушение правил вреда не принесет. Возможно, это так в теории, но это программное обеспечение. Реальность другая. Каждому хорошему продакт менеджеру нужен свой чек лист, когда говорить «да», а когда «нет». И следовать ему без исключений.

Продукты раздуваются по одному неверному решению за раз. Не существует одного выбора, в результате которого появится сообщение: «Ваш продукт теперь слишком сложный». Вы никогда не получите email: «В прошлый вторник, в 14:03 произошел подрыв вашего продукта». Раздутие, сложность и подрыв можно увидеть только в зеркало заднего вида, поэтому всегда думайте об этом риске, говоря «да».

## Функционал и «зависть к физике»

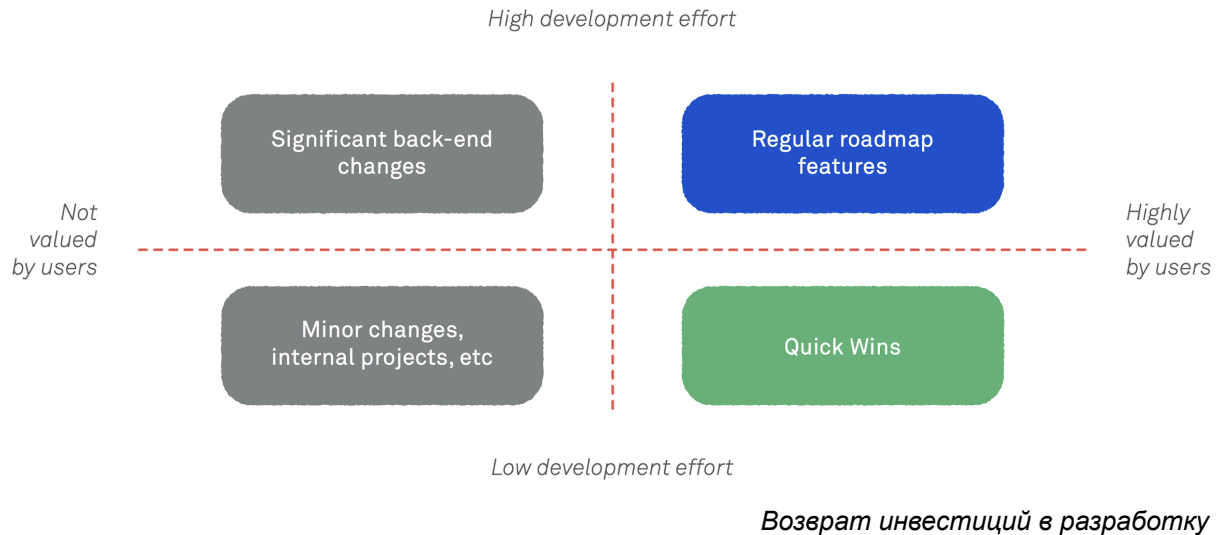


*Каждому понравилось бы, если бы это было правдой*

«[Зависть к физике](#)» термин, использующийся для критики теорий в программном обеспечении и гуманитарных науках, которые пытаются быть более математически строгими, какой стала в свое время физика.

В нашем случае, этим термином можно описать желание дизайнеров, разработчиков и продакт менеджеров, создать законы в системе, которая попросту им не подчиняется. Каждому понравится, если возврат инвестиций будет пропорциональным затраченному времени. Планирование продукта стало бы таким легким. Но проблема в том, что вы можете потратить 5 недель, работая над «убойной» функцией, чтобы потом увидеть как ее проигнорировали пользователи. И в то же самое время, добавить один звуковой эффект, говорящий «привет» или сделать мигающую иконку в браузере, как вдруг каждый начнет говорить о вас. Это не рационально, но когда оно было таким?

Вместо того, чтобы завидовать наукам, в которых нет такой иррациональности, лучше посмотрим, как можно это использовать. В следующий раз, когда будете планировать работу на несколько месяцев, нанесите ваши функции на следующую диаграмму...



Все что находится в правом нижнем углу — особенное. Это функции, где возврат сильно превышает усилия. Некоторые примеры, которые мне доводилось видеть:

- ❑ Изменить текст сообщений в приложении, чтобы сделать его более личным, смешным, точным или полезным.
- ❑ Добавить пару комбинаций клавиш, чтобы опытные пользователи работали более продуктивно.
- ❑ Запоминать последний активный проект, и открывать его при следующем входе пользователя.
- ❑ Добавить дополнительную информацию и ссылки на действия в почтовых уведомлениях.

## Когда использовать быстрые победы

Клиенты не оценят все что вы делаете в продукте. Некоторые необходимые задачи, такие как бэкапы, рефакторинг, оптимизация, улучшение инфраструктуры не дают немедленный эффект. Но это не значит, что их можно игнорировать. Секрет в таком планировании дорожной карты, чтобы не было долгих промежутков времени, когда клиентам кажется, что приложение не обновляется.



Это тот случай, когда помогут быстрые победы. Вы можете распланировать быстрые победы так, чтобы покрыть промежутки времени, когда долго нет видимых изменений. Таким образом, клиенты видят, что продукт постоянно развивается. А вы в это время занимаетесь большими функциями или проблемами бэкенда. Ничего нет хуже, если пользователи видят появляющуюся «паутину» в приложении.

## Выпуск новых функций

Релизы либо получаются либо ломаются, и их очень трудно сделать хорошо. В двойне трудно, если вы меняете привычные процессы пользователей. Для многих B2B приложений — это чья-то работа использовать приложение. Существуют тысячи людей, чья работа использовать Интерком. Поэтому, когда мы планируем изменения в поведении, или добавляем новое, мы относимся к этому не поверхностно.

Для этого у нас есть процесс прогрессивных релизов, состоящих из нескольких шагов, когда мы включаем новые функции лишь для части пользователей. Ваш процесс может отличаться, но для нас этот подход себя хорошо зарекомендовал.

## Тестирование командой

Первый шаг, выпустить релиз для работы с реальными данными, но сделать его доступным лишь для команды, которая создала его. Это позволяет избежать случая, когда слишком много людей сообщают об одних и тех же ошибках. Сама команда знает свой релиз, и могут сказать, это ошибка или просто , что-то не готово.

## Тестирование компанией

Когда команда уверена, что функция готова, они делают ее доступной для всей компании, которая немедленно включает ее в свой процесс работы. Мы не делаем специальных пояснений или извинений. Функция выпускается так, как она была бы выпущена для клиентов. Это позволяет увидеть, что непонятно пользователям. Если уж Интеркомовцы не понимают как пользоваться, какие шансы, что клиенты поймут?

## Ограниченный бета релиз

Первый публичный релиз делаем для менее 10% пользователей. Мы поддерживаем группу «Доверенных тестировщиков» для этих целей. Мы объявляем релиз как ограниченная бета, и просим прислать отзыв, когда они попользовались функцией. Последнее, очень важно запомнить. Отзыв о том, как работает функция может быть только от людей, которые воспользовались ею. Не от тех кто потестировал или поигрался с ней. Джейк Кнапп в статье [«Реакция важнее чем отзыв»](#), привел пример, когда человек, пытаясь быть полезным придумывает улучшения, хотя его реакция была другой. Спекуляциям и дружеским отношениям нет места в разработке продуктов.

### На этом этапе мы проверяем:

- ☐ Находимость — находят ли люди функцию?
- ☐ Вовлечение — пользуются ли они функцией?
- ☐ Принятие — стала ли функция частью их процесса?
- ☐ Сценарии — как функция используется? Какие сценарии популярные?
- ☐ Барьеры — Кто не пользуется? Почему? Что мешает?

Во время ограниченного бета релиза вы можете проверить, что пользователи находят функцию, отшлифовать маркетинговые сообщения, чтобы они

содержали подлинную ценность глазами клиентов, и убрать барьеры использования и принятия.

## Публичный релиз

В этот момент функция доступна всем пользователям, у кого подходящий тарифный план. Теперь самое время подумать, как сообщить им о релизе. Обычно не все функции доступны для всех пользователей – работает обычная дискриминация по цене, поэтому как минимум вы разделяете сообщение на «имеют» и «не имеют» доступ.

Тем, кто имеет доступ к функции, нужно продать ее. Анонсы функций часто сосредотачиваются на том «что мы сделали» вместо того «что вы можете сделать с помощью функции». Если хотите, чтобы люди пользовались продуктом, вдохновляйте их, показывая, что они могут достигнуть.

Тем, кому недоступна функция, нужно продать ее как повод перейти на другой тариф. Это можно сделать при помощи бесплатного пробного периода, например, «Попробуйте премиум тариф в течение одного месяца бесплатно».

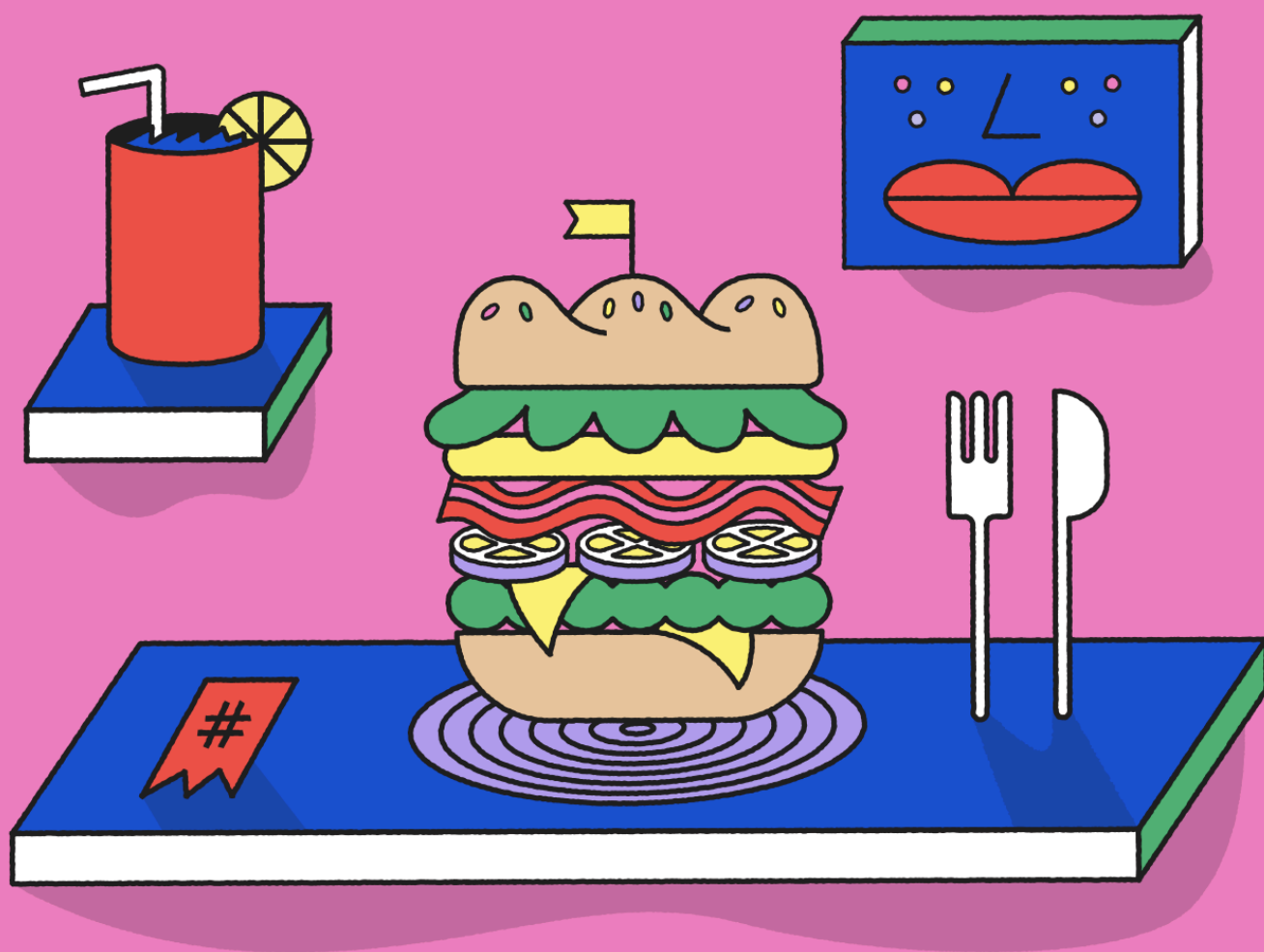
## Расписание сообщения о выпуске

Как только функция была выпущена, легко забыть о ней и перейти к удовольствию разработки следующей. Но функции без вовлечения — это семена для раздутых приложений. При выпуске каждой значительной функции вам необходим план для тех кто будет использовать ее и для тех кто нет.

Когда функция хорошо используется, вы собираете сценарии использования для будущих маркетинговых материалов. В зависимости от ваших планов, вы можете собрать еще и отзывы.

Когда люди не пользуются функцией, вы вдохновляете их, обучаете, вовлекаете. Единственный приемлемый результат в этом случае — они начинают использовать или говорят почему не могут.

Не стоит гордиться большим продуктом с множеством функций. Чем больше плоскостей соприкосновения имеет продукт, тем важнее гоняться за вовлечением и учиться на своих ошибках. Особенно это важно, когда вы молоды и неопытны. Вы можете выпускать новую функцию каждую неделю, но лучше направить эту энергию на увеличение использования продукта, а не на рост продукта.



## Глава 4. Что делать, чтобы функцией пользовались

Решить какие функции нужно добавить и, затем, реализовать их — лишь начало дела. Новым функциям это просто программный код, который собирает

виртуальную пыль, пока ими не начали пользоваться клиенты. Как говорилось в первой главе, если функцией не пользуются — время принять трудное решение: удалить или сделать ее лучше. Не приняли решение, и вы рискуете отдать клиентов конкуренту, у которого продукт лучше сфокусирован и работает эффективнее.

Успешный выпуск функции требует тех же навыков, что и при запуске продукта. Разница в том, что нужно учесть все вопросы с совместимостью с предыдущими решениями и успокоить существующих клиентов. Это не просто.

Большинство новых функций проваливаются. Вы просто не замечаете этого. В этом и проблема. «Новые улучшения» сидят неоцененные, не используемые, и в конце концов, забываются. Добро пожаловать в программное обеспечение. Улучшения даются тяжело.

Вы тратите месяцы на исследования. Анализируете демографию, метрики, этнографию, аналитику, психографию, и что-то там еще. Дни, недели, месяцы встречаетесь с клиентами. Встречаетесь с их родителями, кормите их собак, кормите собак их родителей. Вы делаете все, чтобы понять, что хотят клиенты. Не то что они говорят. Не то что они думают или просят. А то что они действительно хотят. И, затем, вы идете и создаете это!

И оно все равно проваливается. Это жесткая игра.

Почему новый функционал обычно проваливается

Дело в том, что функцию, просто, могут не найти. Об этом часто забывают в спешке подготовки к релизу. Речь не о выкладывании кода на сервера, или о галочке в дорожной карте. Речь о том, чтобы вашим продуктом пользовались. Поэтому в следующий раз перед выпуском новой функции задайте себе следующие вопросы...

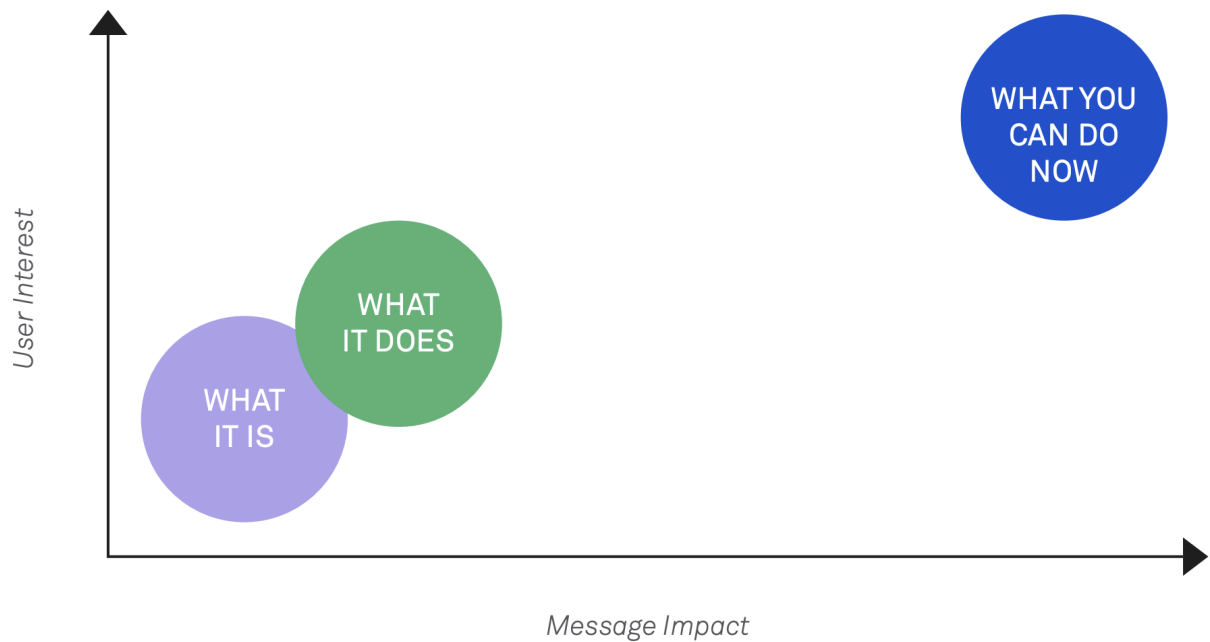
1. Каждый ли может легко найти и понять эту функцию?

**Интересный факт:** когда Microsoft спросила пользователей, что бы они хотели добавить в Office. Оказалось, что 90% функций, которые попросили пользователи уже существовали. Microsoft предположила, что это была проблема осведомленности, несмотря на запуск «[ленты инструментов](#)», которая якобы решала эту проблему. Дело в том, что лента показывала все функции, а значит ни одной. Как я уже сказал, это не просто.

Когда вы разрабатываете первую версию продукта, вы стремитесь, чтобы он был законченным. Поэтому вы не планируете расширение и не оставляете места для будущих функций. Когда вы, все же, добавляете новые функции, вы спешно освобождаете место, и пользователям сложно найти их. Если постоянно возникают вопросы где находится функция, у вас проблемы с обнаружением. В результате, выпуски новых функций будут проваливаться, или как минимум, требовать обучения, чтобы найти и принять их.

2. Вы показываете пользователям, что вы сделали или, что они могут сделать?

Если сказать пользователям, что-то вроде, «переписали с нуля», «на базе HTML5», «адаптивный дизайн», для них это не будет иметь смысла, кроме случая когда вы продаете разработчикам. Никого не интересует, что вы сделали или как вы это сделали. Ваших клиентов заботит, что они могут сделать.



Влияние анонсов функционала

Сосредоточьтесь в вашем сообщении на том, что пользователи могут достичь, благодаря этой функции, и вы получите их внимание.

### 3. Ваши анонсы в контексте?

Новые функции, особенно небольшие дополнения, появляются в продукте без особого контекста. Ваша цель никогда не должны быть «просто выпустить». Цель должна быть — «чтобы пользовались». Сообщения по почте неудачный вариант для таких анонсов, потому что они часто приходят в неподходящее время и в неверном контексте. Подходящее время для анонса улучшений, это когда пользователь находится в приложении, и готов использовать улучшение. Для этого идеально подходят анонсы внутри приложения, которые срабатывают по определенным правилам и событиям.

### 4. Как завтрашние новые пользователи узнают об этом?

По мере роста продукта, он переходит черту, когда его нельзя изучить за один присест. Это не проблема, потому что не каждая функция нужна сразу.



Функции актуальны только когда они решают проблемы. Другими словами, вам не нужны метки для файлов до того как файлы загружены. Вам не нужно слияние меток до того как у вас будет их много. Поэтому, сообщать новым пользователям как делать слияние меток — пример сообщения, показанного не вовремя.

Чтобы научить пользователей как пользоваться функцией в нужное время, есть смысл использовать направленные сообщения. Например, мы рассказываем о сохраненных ответах в Интерком, после того как было отправлено несколько ответов. Мы рассказываем о горячих клавишах когда продуктом пользовались длительное время. Таким образом сообщения приобретают смысл.

#### 5. Планируете ли получить обратную связь от пользователей и не-пользователей?

Как только функция выпущена, вам необходимо получить обратную связь от тех, кто пользуется ею, чтобы понять как, почему и когда ею пользуются. И, затем, найти способы увеличения использования. Вот несколько вопросов, которые я задавал клиентам о новой функции:

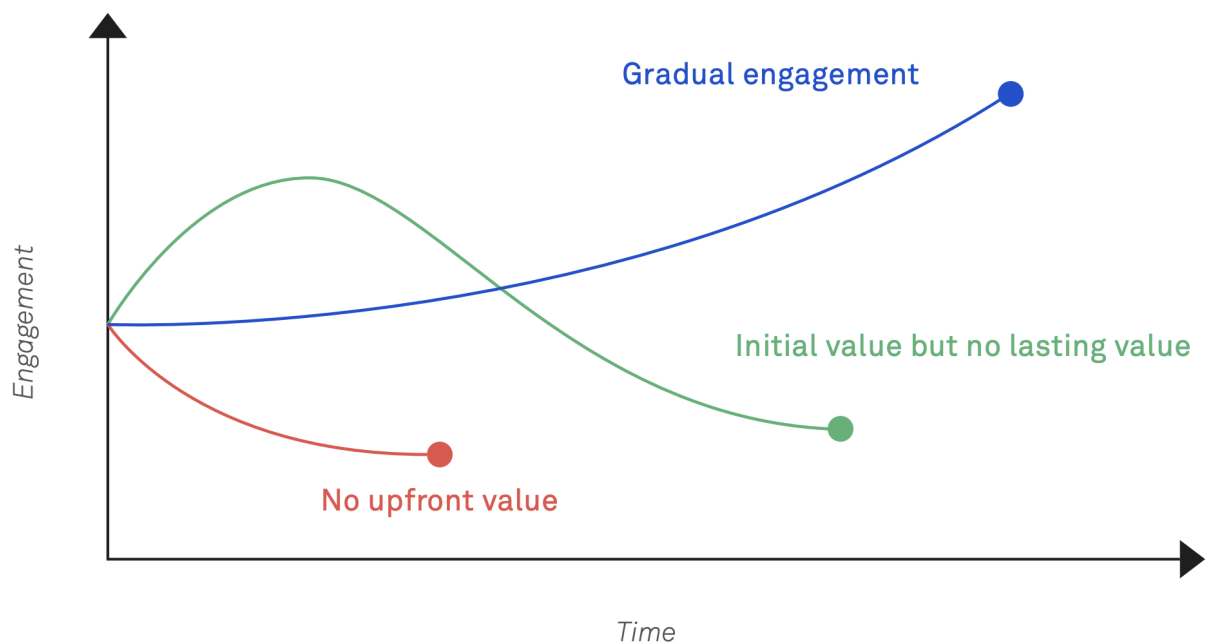
- ☐ Когда вы обнаружили ее? Что вы подумали? Начали ли вы использовать ее сразу? Что привлекло ваше внимание?
- ☐ Требовалось ли вам читать документацию? Была ли она достаточной?
- ☐ Были ли барьеры в использовании функции?
- ☐ Есть ли время в течение вашего рабочего дня, когда вы хотели бы использовать ее, но не можете?
- ☐ Говорили ли вы кому-нибудь для чего вы используете ее? Что они сказали?

Одинаково важно, если не более, получить обратную связь от тех, кто не использует функцию, и понять, что мешает пользоваться. Часто, вы обнаружите препятствия, которые легко устранить, например: «Я все время забываю попробовать функцию», «Я не знаю, пользуется ли функцией

кто-нибудь еще в нашей компании», «Мне потребуется получить CSV файл с данными», и т. п. Все это решаемые проблемы, как только вы их поймете.

Ваш процесс разработки должен допускать, что вы можете сделать что-то не так. Так много дорожных карт и спринтов игнорируют этот факт. Дизайнеры и продакт менеджеры ошибаются все время. И когда это случается, у них есть выбор: исправить или проигнорировать и перейти к следующей функции. Выберите второй вариант достаточное количество раз и вы тоже будете рассказывать, что 90% просьб от пользователей уже есть в вашем приложении. Надеюсь вы любите «ленты инструментов»!

## Увеличение вовлеченности пользователей



*Виды вовлеченности с течением времени*

Определение «вовлеченный пользователь» варьируется от продукта к продукту. Для приложения списка дел, вовлеченный пользователь должен заходить в приложение каждый день, чтобы отметить выполненные задачи. В тоже время для приложения создания счетов-фактур, вовлеченный пользователь может входить лишь раз в месяц. Не существует единого точного определения вовлечения для всех продуктов.

В отличие от таких метрик как: просмотры страниц, количество посетителей, вернувшиеся посетители и конверсии, не существует аналитического приложения, которое сразу могло бы показать то, что вам нужно. Но не спешите игнорировать вовлеченность.

Google+ заявлял, что у них 170,000,000 пользователей. Это число выглядит впечатляюще в новостях. Но при этом игнорируется реальная проблема. Почти никто из этих пользователей не вовлечен. Это просто люди, которые кликнули на ссылку «ОК» когда перед ними появился Google+. Это напоминает как новостные сайты накручивают свои просмотры, используя слайд шоу или многостраничные статьи. В какой-то момент вы должны разобраться кого вы обманываете.

Вовлеченность лишь небольшая часть пазла. Ее часто игнорируют, хотя здесь можно получить много быстрых побед. Вот три способа увеличить вовлеченность пользователей.

#### 1. Произвести хорошее первое впечатление

Каждый день потенциальные клиенты видят ваш интерфейс в первый раз. Об этом часто забывают, особенно команды, которые разрабатывают дизайн для себя. У первого экрана, который видит пользователь, есть три важные задачи:

- ☐ Объяснить как работает приложение.
- ☐ Вдохновить начать пользоваться.
- ☐ Дать знать как получить помощь если потребуется.

Веб приложения, которые не делают ничего из вышеперечисленного получают то для чего они спроектированы — пользователи уходят и никогда не возвращаются.

Существует много замечательных способов поприветствовать пользователя в приложении и помочь сориентироваться. Заготовки документов, пошаговое руководство, вымышленные данные и т.д. В качестве первого шага, мы советуем клиентам Интерком отправить сообщение с приветствием каждому вновь зарегистрировавшемуся пользователю их сайта. Это хорошо работает как начало беседы, что в свою очередь увеличивает конверсию.

Простое общение с вашими пользователями — отличный способ, чтобы они задавали больше вопросов, пробовали новые функции и осматривались. Включаясь в диалог они с большей вероятностью скажут что-то вроде «Как мне отправить email неактивным пользователям» или «Как использовать метки?».

В лучшем случае вы получите больше клиентов, в худшем — узнаете чего не хватает или, что не понятно в приложении.

## 2. Постепенно раскрывайте детали продукта

Каждое стоящее приложение содержит функции, которые не нужны сразу. Это может включать уведомления по email, интеграции со сторонними приложениями, горячие клавиши.

Эти удобства нельзя использовать сразу. Кого волнует экспорт данных до того как появились данные или горячие клавиши до того как начали пользоваться приложением?

Многие владельцы продуктов стремятся показать эти удобства при помощи невовремя отправленных писем, документации или часто задаваемых вопросов. Ни один из этих подходов не работает достаточно хорошо.

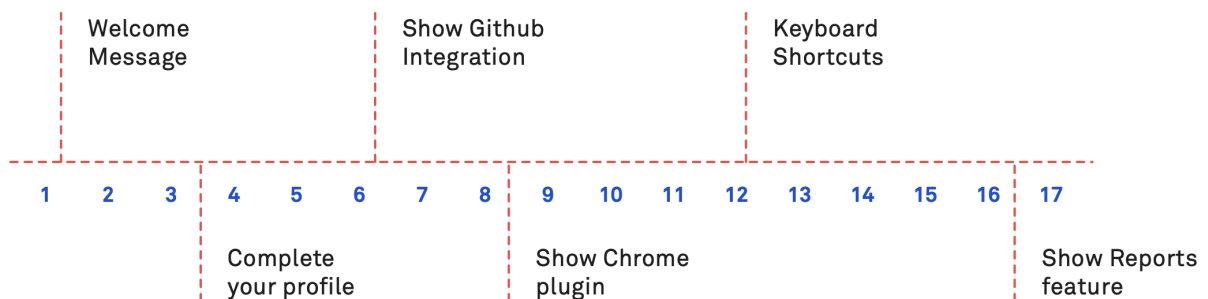
Письма приходят не в контексте и не вовремя, и с большей вероятностью отвлекают пользователя чем интересуют его. Реакция — поместить письмо в архив и перестать читать последующие сообщения.

Показывать функции при помощи часто задаваемых вопросов или документации означает, что их увидят только если у пользователя проблема и он обратился к документации. Это не идеальное время отвлекать его с другими функциями.

Мы рекомендуем нашим клиентам создавать расписание по отправке сообщений, чтобы постепенно показывать некоторые функции. Когда вы хорошо понимаете ваших пользователей, вы сможете решить какие второстепенные функции радуют и в какой момент они полезны.

После выделения таких функций, это вопрос своевременных сообщений. Интерком позволяет регулировать отправку в течение серии сессий. В результате, каждый раз, когда пользователь входит в приложение, вы показываете ему все больше и больше поводов остаться с вами.

Ниже пример расписания для вовлечения от одного из наших клиентов:



*Расписание отправки сообщений по количеству входов в приложение*

Всегда, в каждом сообщении пишите, что вы готовы помочь если есть вопросы. Это ключ к получению обратной связи, который помогает улучшать процедуру вовлечения.

### 3. Анонс функций и улучшений прямо в приложении

Пользователи не замечают когда ваш процесс разработки замедляется. Они заходят в приложение, чтобы пользоваться продуктом, а не отслеживать вашу разработку. Однако, если в продукте ничего не меняется долгое время, их легко отвлечь новостью, о выпуске новой функции конкурентом, полезной или пустячной.

Единственное, когда пользователей интересуют функции — когда они находятся в приложении. Это как раз то место, где их нужно анонсировать.

Мы замечали десятикратное увеличение в коммуникации с пользователями когда использовали сообщения в приложении по сравнению с электронной почтой. При этом были еще плюсы. Например, когда мы сообщили о функции поделиться расположением пользователей на карте, наши клиенты сразу кликнули по ссылке и начали делиться картами. Я не представляю, чтобы от email был бы подобный эффект.

## Создание расписания отправки сообщений

**Сценарий:** Вы продакт менеджер, который хочет увеличить вовлеченность пользователей, когда они находятся вне офиса. Для этого вы разработали мобильную версию вашего приложения. Теперь вам нужно рассказать пользователям о нем.

Слабый способ это разослать email всем зарегистрированным пользователям, прямо сейчас, со ссылкой на мобильное приложение. Что будет дальше?

Некоторые клиенты получают его когда находятся в офисе. Кто-то получит пользуясь вашим веб-приложением на телефоне. Получат и те, кто не пользовался вашим продуктом несколько лет. Кто-то в 4 утра, другие в 10 утра.

Ваша аналитика покажет результаты, что 4.43% людей кликнули по ссылке в письме. И вы будете разочарованы, потому что рассылка была большим и стрессовым проектом.

Можно ли сделать лучше?

## Альтернативные подходы

Вы можете разделить клиентов на две группы: кто заходил в приложение с телефона и кто нет. Это позволит создать подходящее сообщение для каждой группы.

Вы можете создать постоянное сообщение в веб-приложении для тех кто заходит с мобильного телефона, чтобы они знали о мобильное приложение. Вы можете отправить email всем кто вышел из приложения, потому что они еще помнят о вашем приложении. Вы можете создать забавное сообщение, с шутками о конкурентах и включить в него смешные картинки. Благодаря всему этому, люди начинают говорить о вас.

Вы можете перенять у Slack их стиль сообщений о релизах. Эти сообщения интересно и смешно читать. Например:.

«**Исправлено:** Мы исправили уязвимость, благодаря которой Gif можно было произносить как «Жиф». **Исправлено:** Когда клавиатуру убираешь свайпом вниз, окно с сообщениями вдруг прокручивалось назад вверх, это было забавно, но не то что нам нужно.»

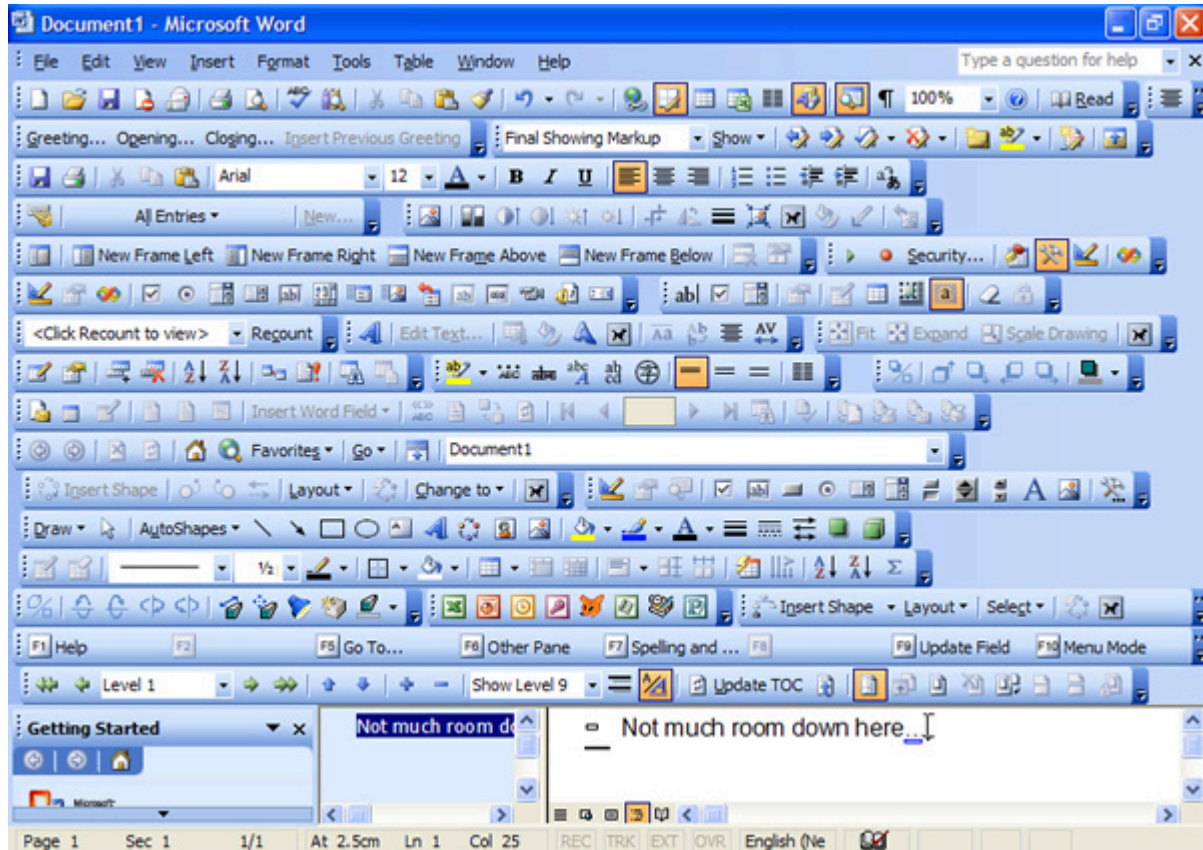
Все эти варианты выходят за рамки «Давайте отправим всем письмо», который выбирают многие компании.

## Не стесняйтесь удалять функции

Когда вы думаете о перегруженных функциями продуктах, что приходит на ум? Бесконечные панели инструментов, настройки и опции, верно?

Было время, когда ни один разговор о пользовательском интерфейсе или стратегии продукта не обходился без упоминания скриншота, на котором был

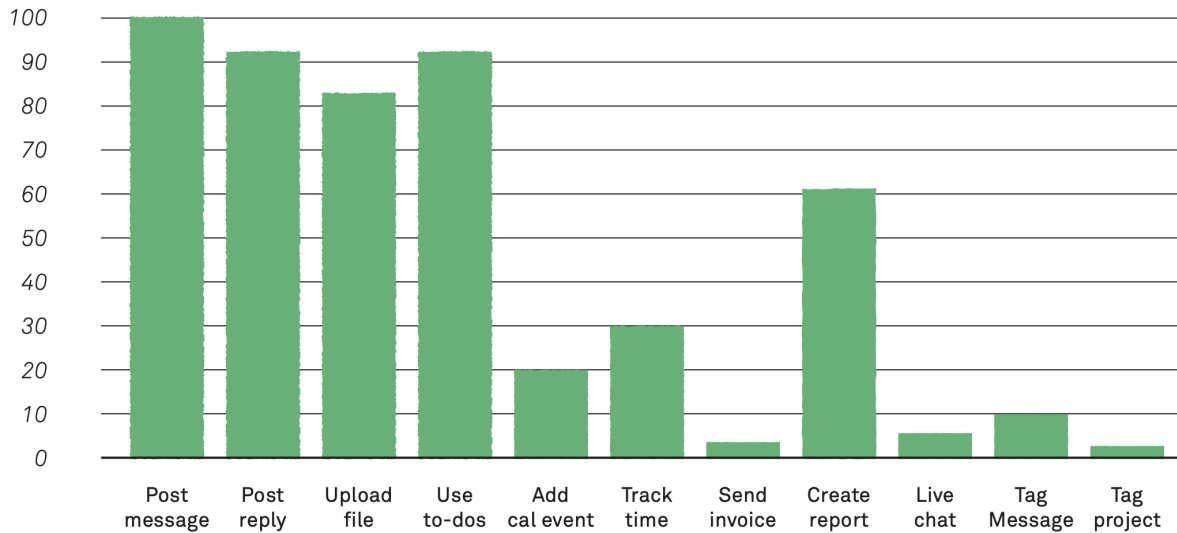
изображен Microsoft Word со всеми панелями инструментов, оставляя лишь небольшое место внизу для документа. Это отличный пример того, как важна простота.



Продукты наподобие этого показывают, что перегруженность функциями обходится недешево, вот почему вам надо легко расставаться с функциями. Пользователи просто не в состоянии освоить все эти функции. Проблема в том, что скриншоты, наподобие Microsoft Word не говорят, что делать дальше. Конечно, здесь слишком много ненужного, но что делать с этим?

Чтобы решить проблему перегруженности функциями нужно определить какими из них пользуются большинство клиентов, а какими нет. Снова время для аудита использования функций, о котором рассказывалось в первой главе.

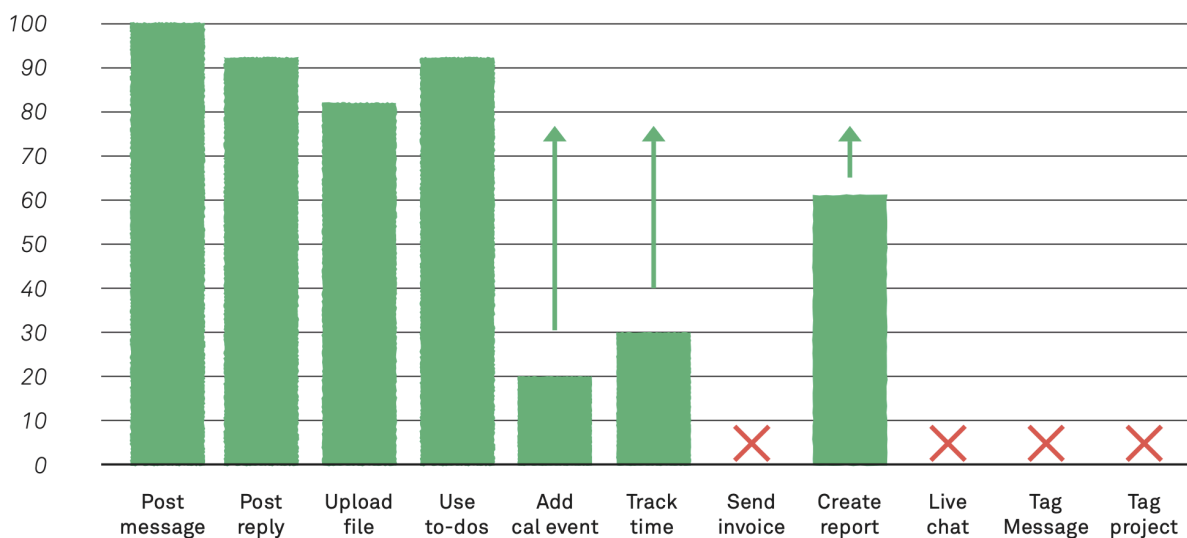




*Выполните повторный аудит функционала*

Если у вас длинный хвост малоиспользуемых функций — вы сбились с курса либо ваша стратегия «создавать все, что попросят». Второе может звучать странно, но продукты наподобие Microsoft Word хорошо справились с этим планом. Но вы вероятно не создаете Microsoft Word. Если вы хотите создать хорошо определенный продукт, вам нужно без сожаления удалять функции.

## Действуйте



Функции не принимают по разным причинам: пользователи не видят пользы в них, они слишком сложные, они спрятаны где-то в продукте, много других. Вот почему важно говорить «нет». Когда вы видите, что лишь 8% пользователей используют ее, нужно принять решение: удалить или сохранить.

Удаляя функцию вы можете скрыть ее от новых пользователей, а существующим — отправить сообщение, объяснив свое решение и давая время, чтобы они могли подготовиться. Или вы просто можете убрать ее из интерфейса и никогда больше не упоминать о ней. Оба варианта работают, и различаются количеством дополнительной работы. Чем больше пользователей, тем аккуратнее должны быть действия по удалению.

Сохранение функции, которую мало используют, требует общения. Вам необходимо выделить пользователей, которые пользуются ею и выяснить почему. Найти пользователей, которые не пользуются и, также, выяснить почему. В этом может помочь такой инструмент как Интерком, который позволяет таргетировать пользователей и начать общение в считанные секунды.

Бывает, достаточно анонсировать функцию в приложении. У некоторых наших клиентов принятие функций возрастало на 30% после отправки сообщения в нужный момент работы в приложении. Также у нас есть клиенты, которые получили достаточно отзывов от их пользователей, чтобы стало ясно, что функция не нужна, и ее удаление будет безболезненным.

Хорошие владельцы продуктов допускают лишь несколько неудачных функций, а великие — удаляют все незамедлительно.

## Заключение: в процессе

Нам пришлось принимать сложные решения о том, какие мысли и части включить в эту книгу. Мы старались сфокусироваться на статьях, которые были

бы полезными и профессионалам и тем, кто недавно пришел в продакт менеджмент. Наше видение о том, как создавать продукты и управлять ими все еще в работе и это наша первая попытка обобщения материалов.

Если эта книга была полезной вам заходите на наш блог [Внутри Интерком](#), где мы регулярно делимся мыслями о продакт менеджменте и связанных темах, таких как дизайн, бизнес стартапов и успехи клиентов.

У нас нет всех ответов, но растет приличный список вопросов.

## Спасибо, что прочитали эту книгу

Мы хотели бы, чтобы вы поделились найденной мудростью с друзьями (ссылки на английскую версию книги):

[Facebook](#) | [Twitter](#) | [LinkedIn](#)