

Глава 3

Ценность

Бережливые решения

Свою книгу *Lean Solutions* (“Бережливые решения”) авторы Джеймс Вумэк (James Womack) и Дениэл Джонс (Daniel Jones) начинают с признания, что им самим приходится быть потребителями (или клиентами), которым необходимо решить какие-то проблемы, вследствие чего они часто бывают раздражены из-за множества трудностей, с которыми сегодня неизбежно связан статус клиента. С высоты своего положения (как потребителей) каждый из них хочет сказать следующее тем, кто предоставляет им товары и услуги.¹

- Решите мою проблему кардинально.
- Не заставляйте меня тратить время.
- Дайте мне именно то, что я хочу.
- Предоставьте то, что мне требуется, именно там, где мне это нужно.
- Обеспечьте то, в чем я нуждаюсь, именно тогда, когда мне это нужно.
- Сократите до минимума число решений, которые я должен принять, чтобы решить свою проблему.

Давайте познакомимся с компанией, которая согласилась с этими пожеланиями клиентов, кажется, еще несколько лет назад.

Google

Google, возможно, единственная компания в мире, которая декларирует цель добиться, чтобы пользователи покидали их веб-сайт как можно скорее².

В 1999 году компания Google вышла на переполненный рынок поисковых средств с амбициозной целью сделать доступной для всех информацию всего мира. Компания решила, что для достижения поставленной цели необходимо сосредоточиться на двух моментах.

¹ Womack J., Jones D. *Lean Solutions: How Companies and Customers Can Create Value and Wealth Together*, Free Press, 2005, p. 15.

² С веб-сайта Google, www.google.com/corporate/tenthings.html.

- Предоставлять более существенные результаты поиска, чем конкуренты.
- Сделать так, чтобы у тех, кто пользуется данным поисковым средством, оставались приятные впечатления.

Идея сработала. В своем последнем выпуске за 1999 год журнал *PC Magazine* присудил поисковому средству Google премию за техническое совершенство среди веб-приложений. Как отмечалось в *PC Magazine*: “Не требуется много раз обращаться к услугам Google, чтобы оценить преимущества этого поискового средства. И как можно не оценить средство, которое постоянно предоставляет хорошие результаты.”³ Премия привлекла внимание к Google, и в результате данная компания начала головокружительный подъем.

Корпоративная философия Google, опубликованная на веб-сайте компании, начинается со следующих четырех пунктов.⁴

1. Основное внимание — потребностям пользователя, а все остальное приложится.
2. Лучше сделать что-то одно, но по-настоящему качественно.
3. Демократия возможна и в веб.
4. Лучше быстро, чем медленно.

В начале своего существования компании Google пришлось прилагать значительные усилия, чтобы не изменить своим принципам. Всплывающие окна и реклама, столь популярные на других веб-сайтах, были строго запрещены в Google, поскольку они вызывали недовольство пользователей. Компания всегда очень ревностно относилась к быстрой работе своих средств. Любая осязаемая задержка с выдачей результатов рассматривалась как “растранивание” времени клиента. В то время как большинство конкурентов создавали порталы, начальная страница Google всегда была предельно простой: единственный логотип, около тридцати слов текста, одно поле ввода и пара кнопок. В компании полагали неправильным отвлекать внимание пользователей излишествами.

В конце концов предоставляемые Google результаты поиска приобрели качество и полноту и с течением времени становились все лучше. Основатели компании создали уникальный метод классификации веб-страниц, представленных в результатах поиска, — вычислялось число “голосов”, поданных за ту или иную страницу, т.е. число и значимость ссылок на нее. Страница с большим числом ссылок получала преимущество. В дальнейшем компания Google продолжала совершенствовать свой метод классификации веб-страниц.

В процессе создания своего продукта (в данном случае поискового средства) Google применяла все те же четыре принципа.

1. **Ценность: основное внимание — потребностям пользователя, а все остальное приложится.**

Коллективы разработчиков имели систему поощрений, стимулирующую создание продуктов, ориентированных на интересы пользователей. При этом предполагалось, что, после того как продукт приобретет популярность, руководство компании сумеет извлечь из него прибыль.⁵

³ PC Magazine, December 14, 1999, p. 104.

⁴ С веб-сайта Google, www.google.com/corporate/tenthings.html.

⁵ Vise D. A Google's Missing Piece, Washington Post, February 10, 2005., p. E05.

2. Совершенство: лучше сделать что-то одно, но по-настоящему качественно.

Компания Google, так же, как и ее знаменитое поисковое средство, великолепна в своем деле. Высокий технический уровень коллектива позволяет экспериментировать во многих направлениях одновременно.⁶

3. Демократия: и в веб возможна демократия.

Новые идеи тут же получают поддержку в Google, привлекая критическую массу людей, стремящихся работать над ними. Сам факт, что продукт представлен на сайте Google Labs, популяризует его и привлекает к нему внимание пользователей.

4. Скорость: лучше быстро, чем медленно.

То, на что у других компаний уходят годы, в Google осуществляется в течение нескольких недель или дней.⁷

Обратите внимание, что Google, к своему спектру услуг, добавила карты. В марте 2004 года была выпущена бета-версия Google Local — средства, позволяющего ограничить поиск определенной географической областью. В октябре того же года Google приобрела компанию Keyhole, что предоставило ей доступ к изображениям земной поверхности, получаемым с помощью спутников. В феврале 2005 года в Google Local были добавлены карты. А изображения из космоса появились в наборе функциональных возможностей Google в апреле 2005 года — всего через шесть месяцев после приобретения Keyhole! И всего двумя месяцами позже было выпущено Google Earth — средство, предоставляющее доступ к изображениям (земной поверхности) из космоса, картам и локальному поиску с помощью инновационного интерфейса на рабочем столе, имитирующего глобус.

Компания Google сосредоточилась на своей стратегической задаче: сделать доступной и полезной для всех информацию всего мира. Для ее решения Google создает как программные средства, используемые внутри компании, так и веб-продукты, которые включают изощренные алгоритмы, сложную аппаратуру и тщательно разработанное программное обеспечение. Продукты и средства разрабатываются быстро и поэтапно небольшими коллективами. При этом поощряется выпуск программных продуктов до того, как они будут полностью завершены — с целью создания обратной связи с пользователями. Программное обеспечение компании Google, которое неизменно восхищает пользователей, славится элегантным дизайном и высоким качеством.

По всем меркам Google — очень успешная организация, и, что более важно, ее продукты непрерывно совершенствуются в соответствии с тем, как изменяется мир. Google постоянно представляет новые продукты, несмотря на то, что эта компания не имеет долговременного плана создания новых продуктов. Компания определяет целесообразность создания того или иного продукта приблизительно так же, как она классифицирует веб-страницы в результатах поиска: приоритет определяется энтузиазмом коллектива разработчиков и интересом пользователей.

Авторы полагают, что компания Google добилась таких больших успехов благодаря тому, что она предоставляет пользователям именно ту информацию, которая им нужна, в нужное время и в нужном месте, без проблем, не заставляя пользователей ждать, и без надоедливой рекламы. Средства Google достаточно просты в использовании, даже для ма-

⁶ Hammonds K.H., *How Google Grows...and Grows...and Grows*, Fast Company, Issue 69, April 2003., p. 74.

⁷ <http://video.google.com/videoplay?docid=-8618166999532839788>.

ленького ребенка. И они позволяют решать проблему комплексно — проверить орфографию, осуществить перевод с одного языка на другой, конвертировать единицы измерения; кроме того, здесь имеются словарь, калькулятор и карты. Словом, все это в сумме полностью соответствует определению “бережливое решение”.

От концепции до прибыли

Давайте присмотримся к графику работы над первым продуктом Google, чтобы познакомиться с функционированием этой компании.

Концепция

В 1996 году Ларри Пейдж (Larry Page) и Сергей Брин (Sergey Brin) объединили усилия для решения интересующей их задачи — поиска информации в Интернете. В то время они были студентами-выпускниками Станфордского университета (специальность — информатика), занимавшимися исследованием проблем поиска данных, поэтому они решили попробовать свои силы в поиске информации в Интернете. Двумя годами позже они опубликовали работу, в которой были изложены подробности концепции, которая позже легла в основу работы поискового средства Google.⁸

Предложить блестящую концепцию (которая может быть результатом или долгих исследований, или озарения) — это только начало. Компании могут создавать условия для появления инноваций, но не существует магической формулы для озарений. Однако обычно озарения недостаточно: “Инновации — это 5 процентов озарения и 95 процентов тяжелого труда”⁹. Проблема большинства компаний в том, что люди “с вдохновением” не имеют времени или возможностей реализовать его. В то же время компании, постоянно работающие над тем, чтобы превратить вдохновение в инновацию, предоставляют время и создают возможности для людей и коллективов с хорошими идеями, чтобы эти идеи смогли созреть и превратиться в концепции.

Осуществимость проекта

Компания Google, подобно Dell, “начиналась” в комнате студенческого общежития. Когда первый интерфейс поискового средства Google появился на веб-сайте Станфордского университета, компьютеры, с которых можно было осуществлять поиск данных в Интернете (с помощью нового средства), располагались в комнате Пейджа. Вскоре Пейдж и Брин оставили учебу ради бизнеса. После этого худшее, что с ними могло бы случиться, это что им придется вернуться в университет, чтобы завершить свое образование. Компания Google была основана в 1998 году, однако почти в течение года продукт новой компании имел статус бета-версии, в то время как Пейдж и Брин исследовали осуществимость проекта. В это время компания начала привлекать внимание средств массовой информации, трафик рос и Пейдж с Брином смогли обеспечить необходимое финансирование. Наконец пометка “бета” была удалена и исследовательский проект стал служить основой реального продукта.

⁸ Brin S., Page L. *The Anatomy of a Large-Scale Hypertextual web Search Engine*, Computer Networks and ISDN Systems, 30(1-7):107-117, April 1998.

⁹ Цитата Томаса Эдисона.

Стадия определения осуществимости в процессе разработки продукта важна, поскольку она предоставляет возможности для экспериментирования. Следует заметить, что стадия определения осуществимости — это не кабинетные (или бумажные) исследования. Концепция — это прекрасно, но ничто не заменит ее проверки в реальных условиях.

Изготовленные вручную образцы

На предприятии по производству видеокассет, где мне пришлось работать, из этапа определения осуществимости проекта мне больше всего запомнились образцы нового продукта (или изделия), которые были произведены в лаборатории вручную с помощью самого примитивного оборудования. Но если не вспоминать о примитивном оборудовании, эти изготовленные вручную образцы были очень близкой имитацией конечного продукта.

Мы израсходовали множество этих образцов, к каждому из которых был приклеен аккуратный ярлык. Мы различным образом испытывали эти образцы — на механическое разрушение, воздействие солнца или влаги или, может быть, отдавали их пользователям для испытания в реальных условиях.

Для меня стадия определения осуществимости связана с большим количеством экспериментов, позволяющих выяснить (в реальных условиях и у реальных пользователей), что в новом продукте хорошо и что плохо. (Мэри Поппендик)

Это также время принятия решений, определяющих конструкцию (или архитектуру) будущей системы, — критически важная задача, которую многие компании игнорируют. Каковы основные свойства бизнес-процесса или ключевых модулей аппаратуры? Какие интерфейсы будут использоваться в системе и как они будут взаимодействовать? Как архитектура программного обеспечения будет поддерживать продукт (или изделие)? Где в системе узкие места и какие меры следует предусмотреть, чтобы избежать проблем? Совершенный проект системы — основа будущего совершенного продукта. Эта задача не должна возлагаться ни на дилетантов, ни на далеких от практики экспертов. Справиться с ней в состоянии только зрелые разработчики, которые знают, что проект системы будет эволюционировать по мере создания продукта, и которые знают, как обеспечить, чтобы эта эволюция принималась во внимание.

Пробная версия

Компания Google начала с системы классификации веб-страниц, основанной на подсчете ссылок на эти страницы. Как только эта система заработала, ученые приступили к сбору данных обо всем: какие результаты были использованы и какие нет, какие фразы были заданы для поиска данных результатов, какие опечатки наиболее типичны, как быстро предоставлялись результаты поиска и т.д. Компания Google непрестанно совершенствовала свою систему классификации страниц, а также аппаратные средства и архитектуру программного обеспечения, основываясь на анализе этих обширных данных. Были разработаны средства, облегчающие разработку приложений для доступа к “безбрежным” информационным структурам компании. Создавалось множество пробных приложений, которые передавались в Google Labs, где пользователи могли проголосовать за тот или иной продукт и помочь совершенствовать их через обратную связь.

Никто не ожидает, что на стадии определения осуществимости проекта появится готовый, сформировавшийся продукт. На этом этапе только определяется, будет ли создаваемый продукт работать. В этот момент реальная разработка продукта только начинается. Правильно построенный процесс разработки состоит из серии проб и ошибок, кото-

рые позволяют усовершенствовать дизайн продукта. Целью этого этапа является не завершить создание продукта, а довести его до состояния, когда он (с минимальным набором функциональных возможностей) может быть выпущен в виде пробной версии для оценки пользователями. При этом предполагается, что после первой пробной версии последуют другие, более полные, и таким образом продукт постепенно обретет контуры.

Что касается стадии испытания пробной версии программного продукта, то здесь имеется большая вероятность выпуска и передачи подобной версии пользователям для испытаний (чем на аналогичной стадии создания аппаратуры), хотя, как будет показано в главе 8, в рамках программы разработки подводных лодок Polaris были созданы несколько функционирующих подлодок в период от трех до восьми лет после начала программы. Однако, даже если пробная версия аппаратуры (или прототип) реально не передается пользователям для испытаний, все равно удачной идеей является планирование создания прототипов, включающих встроенное ПО, так рано в процессе разработки, как это возможно.

Выпуск программного обеспечения (и передача его пользователям для оценки) на стадии создания пробной версии почти всегда оказывает благотворное действие на процесс разработки (если на стадии определения осуществимости проекта были приняты правильные проектные решения для будущей системы). Конечно, выпуск ранней версии программного продукта не всегда целесообразен. Встроенное программное обеспечение, например, не может быть выпущено раньше аппаратуры. Игровые программы также не могут быть выпущены в незаконченном состоянии, поскольку первое впечатление об игре — это почти все, а срок жизни игровых программ очень краток. Однако в большинстве ситуаций все же целесообразно осуществлять ранний выпуск программного обеспечения в виде альфа и бета-версий для его оценки.

Прибыль

Компания Google в конце концов начала зарабатывать много денег — пропорционально ценности, предоставляемой рекламодателям и пользователям. Компании пришлось много экспериментировать и многому научиться, прежде чем было определено, как сделать рекламу неназойливой и такой же уместной, как результаты поиска. Google приходится сталкиваться с конкуренцией, и, чтобы обеспечить постоянные прибыли, компании необходимо неустанно заботиться о том, чтобы пользователи были довольны.

Восхищенные потребители

В 1984 году доктор Нориаки Кано (Noriaki Kano) из Токийского университета опубликовал работу, посвященную “свойству привлекать”, которая явилась вехой в своей области.¹⁰ В этой работе была представлена модель Кано (рис. 3.1) — средство, позволяющее прояснить, как создаются выдающиеся продукты, восхищающие потребителей.

Модель Кано демонстрирует: для того чтобы только “протиснуться в дверь”, необходимо удовлетворить элементарные потребности. После этого открываются две перспективы: 1) увеличивать привлекательность продукта, добавляя в него новые функциональные возможности; и 2) выявить потребности пользователя, о которых он пока сам не до-

¹⁰ Noriaki K., Seraku N., Takahashi F., Tsuji S. *Attractive Quality and Must-Be Quality*, *Hinshitsu*. The Journal of the Japanese Society for Quality Control, April 1984, p. 39–48.

гадывается, и восхитить его, неожиданно удовлетворив эти потребности. Доктор Кано отмечал, что простое улучшение характеристик (или добавление новых функциональных возможностей) обеспечивает лишь линейный рост удовлетворенности продуктом — в этом случае удовлетворение потребителя возрастает пропорционально числу новых функциональных возможностей. Чтобы получить экспоненциальный рост удовлетворенности, необходимо выявить и удовлетворить потребности, на удовлетворение которых потребитель сам пока не рассчитывает. Кано утверждает, что пользующиеся спросом продукты не создаются на основе только пожеланий потребителей. Для этого производителю необходимо глубоко понимать запросы потребителя, обнаружить его скрытые запросы, удовлетворить которые пока еще никто не пытался, и таким образом удивить и восхитить его.

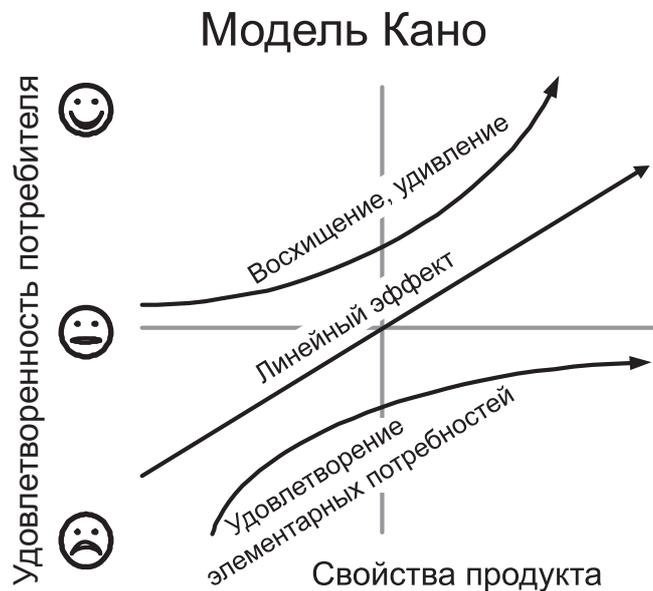


Рис. 3.1. Модель Кано

Компания Google, кажется, очень успешна в том, чтобы восхищать пользователей. Во время путешествий по миру с докладами и курсом лекций мы часто задавали вопрос: “Кто из присутствующих доволен Google?” Почти все поднимали руки, а затем мы узнавали о множестве способов использования продуктов Google. Наш сын регулярно посылает нам ссылки на неизвестные пока новые функциональные возможности. А наша внучка (учащаяся начальной школы) постоянно использует средства Google для выполнения домашних заданий.

Глубокое понимание потребителя

Том¹¹ однажды сказал интервьюеру: “Великолепное программное обеспечение является результатом объединения усилий тех, кто знает соответствующий бизнес (в котором будущее ПО предполагается использовать), и тех, кто разбирается в соответствующей технологии”. Глаза репортера неожиданно зажглись. “Я понял!” — воскликнул он, а затем рассказал следующую историю.

Это было в конце 90-х годов, когда каждой газете вдруг потребовалось иметь свой веб-сайт. Все другие газеты, для создания своих сайтов, обратились к услугам соответствующих фирм. В отличие от них, наше издание наняло программиста, и работать с ним было поручено мне. Сначала было очень трудно уловить, о чем он говорит, однако через некоторое время я начал понимать, чем свой веб-сайт может быть очень полезен для нас. В этой ситуации я представлял бизнес, а программист, о котором шла речь выше, был техническим специалистом, и вместе мы разобрались, что желательно для нашей газеты и что возможно. В конце концов это сотрудничество позволило создать нечто полезное. Наши конкуренты предоставляли доступ к содержимому своих сайтов совершенно бесплатно, в то время как мы за доступ брали все увеличивающуюся плату. И тем не менее очень многие подписчики были готовы раскошелиться. В настоящее время все газеты, аналогичные нашей, испытывают финансовые трудности, а мы даем существенную прибыль.

В мире имеется много программного обеспечения среднего качества, и только иногда что-то по-настоящему инновационное привлекает наше внимание. И мы хотим знать *Как им это удалось?*, или точнее, *Как бы нам сделать то же?*

Фокусироваться на работе

В 1982 году 30-летний Скотт Кук (Scott Cook) задавал себе тот же вопрос.¹² Только что на рынке появился персональный компьютер IBM PC, и Кук был один из многих предпринимателей, задумавших создать программные продукты для продажи владельцам ПК. Кук имел преимущество. До этого он пять лет работал в компании Procter & Gamble, где приобрел опыт, как следует предлагать продукты, которые изменят жизнь людей. Кук сделал то, что на его месте бы сделал любой хороший работник отдела маркетинга в P&G. Он начал с поиска работы (или задачи), которую необходимо было выполнять и для которой пока не существовало никаких подходящих средств. Он обратил внимание на отчаяние своей жены, когда она приводила в порядок семейные финансы, и решил, что соответствующее программное обеспечение существенно облегчило бы задачу. Кук был не одинок в своем открытии. Ко времени, когда он основал компанию Intuit и представил программный продукт Quicken в 1983 году, у него было уже 46 конкурентов.

В P&G Кук усвоил, что для борьбы с конкурентами прежде необходимо до конца понять задачу, стоящую перед потенциальными потребителями, и выяснить, каковы недостатки средств, используемых для ее решения в настоящее время, и что именно означает выполнить данную работу лучше. Он начал с методичного исследования, как люди управляют своими финансами. Он выяснил, что большинство делают при этом только три вещи: они платят по счетам, контролируют свою чековую книжку и периодически суммируют и про-

¹¹ Один из авторов. — *Примеч. ред.*

¹² Taylor S., Schroeder K. *From Inside Intuit*, Harvard Business School Press, 2003.

смотривают свои расходы. И он обнаружил, что люди находят это занятие неприятным и стремятся освободиться от него как можно скорее. Поэтому программа Quicken с самого начала была создана для выполнения трех (и только трех) задач — быстрее, чем их можно было бы выполнить вручную.

Далее Кук обратил внимание, что конкурирующее программное обеспечение по управлению финансами перегружено излишними функциональными возможностями и установка его на компьютере занимает несколько часов. Кук решил создать такую простую программу, которую даже пользователь-новичок смог бы быстро установить на ПК и заполнять чеки уже через пятнадцать минут. Затем он создал систему тестирования простоты использования с тем, чтобы программу Quicken 1.0 можно было тестировать и упрощать до тех пор, пока она не будет соответствовать поставленным целям. Тем временем те же тесты показали, что опытному пользователю ПК необходимо около пяти часов, чтобы выполнить установку и создать первый чек на программном обеспечении конкурентов, перегруженном излишними функциональными возможностями.

Поскольку просто иметь лучшее средство недостаточно, компания Intuit занялась исследованиями в области ценообразования и упаковки. Люди, которым необходимо было выполнить некоторую работу, должны были сразу понять, что это лучшее средство для ее выполнения. Кроме того, Intuit занималась постоянным совершенствованием своей программы, чтобы она оставалась наилучшим средством для выполнения данной работы. Кук полагал, что потребительскую ценность можно создать, если глубоко вникнуть в суть работы, которую необходимо выполнить, и усовершенствовать программное обеспечение настолько, чтобы с его помощью можно было бы выполнять эту работу лучше и быстрее, чем при использовании альтернативных средств.¹³

Мои заказчики

Когда я участвовала в создании системы управления процессом, не возникало вопроса, кто является заказчиками (рядовые рабочие, выполняющие реальную работу), и в чем состоит работа (в производстве ленты для видеокассет). Если наша система не окажется полезна при выполнении данной работы, рабочие проигнорируют новую систему и будут производить ленту без нее. Условия были ясны: если новая система не будет использоваться, вина за это ляжет только на нас — разработчиков: значит, мы не поняли до конца работы, облегчить выполнение которой была призвана наша система. (Мэри Поппендик)

Ориентированная на потребителя организация

Как задумываются и создаются выдающиеся продукты? Книга Кларка и Фуджимото *Product Development Performance*¹⁴ (см. главу 1), появившаяся в 1991 году, рассказывает, что выдающиеся продукты являются результатом хорошо организованного потока информации. То, как потребители воспримут продукт, определяется качеством информационного потока между рынком и коллективом разработчиков. Техническая целостность продукта определяется качеством информационного потока между членами этого коллектива.

¹³ Christensen C., Cook S., Hall T. *Marketing Malpractice: The Cause and the Cure*, Harvard Business Review, December, 2005.

¹⁴ Clark K., Fujimoto T. *Product Development Performance*, Harvard Business School, 1991.

Способствовать этому потоку можно с помощью двух мер: 1) обеспечить руководство и 2) поручить работу многофункциональному коллективу (complete team).

Руководство

Давно замечено, что создание успешного продукта тесно связано с присутствием *чемпиона* (или лидера) — лица, глубоко понимающего суть проблем (или задач), стоящих перед потребителем, и способного предложить решение, которое бы удивило и восхитило этого потребителя. На чемпионе лежит ответственность в принятии ключевых (для создаваемого продукта) решений, а также за результаты этих решений. Вот как об этом говорит Мартин Кэган (Martin Cagan) из компании Silicon Valley Product Group.

Создание выдающегося продукта возможно только, если это возложено на лицо с глубоким сочувствием к потребителю, даром видеть возможное, а также способностью понять, что существенно и что случайно. Это лицо глубоко понимает как потребителя, так и возможности своего коллектива разработчиков. Такое лицо может занимать должность менеджера продукта или быть кем-то другим — от рядового инженера до учредителя компании. Главное, чтобы эту роль взял на себя кто-то, кто не боится ответственности, а также обладающий необходимыми качествами.

Когда приходится наблюдать программу разработки некоторого продукта в бедственном положении, обычно всегда оказывается, что роль чемпиона здесь вакантна. Существует несколько моделей реализации этой роли. Для лучших моделей характерна ясность, на ком лежит ответственность за успех продукта.

Главный инженер

В компании Toyota вся ответственность за экономический успех семейства автомобилей ложится на главного инженера, ответственного за данное семейство. Этот руководитель является высококвалифицированным инженером с глубоким знанием конструкции автомобиля. Кроме того (с помощью небольшого штата) он старается глубоко понять потребителя (для которого предназначена та или иная модель) и что этот потребитель ценит. Подобный руководитель знакомится с исследованиями рынка, общается с дилерами, а также регулярно посещает места, где часто можно встретить потенциальных потребителей. Он получает финансирование и общие указания от вышестоящего менеджмента. Однако в конце концов именно главный инженер лично занимается объединением всей этой информации, решает, что понравится потребителям, создает концепцию будущего продукта и руководит разработкой.¹⁵

Возьмем, например, модель Sienna. Первая версия автомобиля типа minivan (небольшой фургон) компании Toyota продавалась не особенно хорошо. Когда главный инженер Юджи Йокоя (Yuji Yokoya) взялся улучшить модель, он знал, что ему, чтобы справиться с задачей, потребуется не только группа для изучения соответствующего сегмента рынка и информация от потребителей. Поэтому он последовал традиции Toyota “разобраться на месте”. Он проехал на Sienna 53000 миль (85000 километров) через каждый штат в США и Мексике, а также каждую провинцию в Канаде. Обычно он путеше-

¹⁵ Информация для данного абзаца взята из главы 7 диссертации (Sobek D. K. II) “Principles That Shape Product Development Systems: A Toyota-Chrysler Comparison,” диссертация была представлена на соискание ученой степени доктора философии (специальность Industrial and Operations Engineering) в университете Мичигана, 1997г.

ствовал с одним из ключевых членов коллектива разработчиков, включая Джона Джулэ (John Jula), инженера с выдающейся комплекцией, которому предстояло переделать конструкцию сидений. В ходе путешествия Йокоя пришел к пониманию, что должен оценить потребитель: больше пространства, удобные передние сиденья для родителей и задние — для детей. В результате продажи новой модели Sienna в 2004 году удвоились и данная модель оказалась среди лучших.

Мы (авторы. — *Примеч. ред.*) недавно купили минивэн Sienna. Мы всерьез рассматривали покупку других автомобилей, однако нашли, что пространство для ног водителя и переднего пассажира в других аналогичных моделях слишком тесно. Отпуск мы проводим “на колесах”, поэтому удобство передних мест по-настоящему важно для нас. Мы могли согласиться, когда попробовали Sienna, что Джон Джулэ думал о нас при конструировании передних мест этого автомобиля.

Преимущества, связанные с существованием должности главного инженера, состоят в том, что одно лицо берет на себя ответственность за объединение и осмысление множества разнородной информации. Кроме того, дух предпринимательства, возникающий там, где имеется частная собственность, имеет долгую историю создания блестящих инноваций. Люди любят следовать за успешными руководителями, поскольку они помогают делать успешными целые коллективы.

В качестве примера можно также вспомнить, что большинство проектов Open Source¹⁶ имеют “сильного руководителя проекта” (аналог главного инженера).¹⁷ Подобные проекты обычно начинаются с идеи одного индивидуума, и даже если множество добровольцев работают над проектом, весь программный код просматривается и исправляется руководителем или специально отобранными помощниками.

Должность главного инженера — не панацея. Вполне возможна ситуация, когда кто-то, занимающий эту должность, видит свою задачу в том, чтобы “руководить,” а не объединять усилия всей команды. Это не проблема для проектов Open Source, поскольку невозможно для руководителя такого проекта привлечь разработчиков-добровольцев, не являющихся энтузиастами своего дела. Однако это может стать проблемой в компаниях, где должность главного инженера проекта не является хорошей школой для профессионального роста, или в компаниях, где полагают, что главный инженер должен сосредоточиться больше на решении конкретной задачи, а не на объединении информации (о возможных вариантах конструкции и о запросах потребителя).

Многие успешные компании, занятые созданием программного обеспечения, были основаны главными инженерами, в которых технические знания объединялись с ясным видением, в чем нуждается незаполненный рынок. Хорошим примером является eBay¹⁸ (подобные компании, как правило, создают главные инженеры). К сожалению мы выдели множество успешных главных инженеров, которые, основав компанию, по мере ее

¹⁶ Открытое программное обеспечение, т.е. ПО с “открытым” исходным кодом (англ. open source software) — способ, при котором исходный код создаваемых программ открыт, т.е. общедоступен для просмотра и изменения. Это позволяет всем желающим использовать уже созданный код для своих нужд и, возможно, помочь в разработке открытой программы. Более подробно об Open Source можно прочесть в главе 9.

¹⁷ Raymond E. S. *The Cathedral and the Bazaar*, www.catb.org/~esr/writings/cathedral-bazaar/cathedral-bazaar/, 2000.

¹⁸ (читается: “И-Бэй”) — компания, предоставляющая услуги интернет-аукционов (основное поле деятельности), интернет-магазинов, мгновенных платежей.

роста становились посредственными генеральными директорами (СЕО). Они не потрудились вырастить новых главных инженеров, часто передоверяя задачу определения потребительской ценности (customer value) будущего продукта отделу маркетинга. Через пять или десять лет такие компании часто приходят в упадок, по мере того как появляются все больше конкурентов и не оказывается на месте главного инженера, способного предвидеть будущее и нацелить коллектив на создание конкурентоспособных продуктов.

Команда руководителей

Целью создания нового продукта всегда является удовлетворение некоторого спроса, который до этого оставался неудовлетворенным, — либо из-за отсутствия необходимой технологии, либо из-за того, что этот спрос был не распознан, либо из-за какой-то комбинации первых двух причин. В последнем случае подчас лучший способ соединить две идеи состоит в том, чтобы иметь двух руководителей, один из которых обладает глубоким пониманием рынка, а второй — обширными техническими знаниями. В компании Intuit, например, практика объединения эксперта в области маркетинга с техническим специалистом началась с основателей. Скотт Кук определил видение рынка для программы Quicken, в то время как Том Проул (Tom Proulx) руководил техническими разработками.¹⁹

Если в этом уравнении имеется третий важный элемент, — производство, например, — имеет смысл иметь трех руководителей разработки продукта. В компании Honda при создании новых моделей практикуется объединение усилий специалистов по продажам, по производству и в области опытно-конструкторских работ, которые работают как единая команда при принятии ключевых для создаваемого продукта решений. Помимо команды руководителей, Honda также имеет в крупных проектах единого руководителя, роль которого сходна с ролью главного инженера проекта в компании Toyota.

Крайне важно обеспечить качественное руководство в наиболее важных точках процесса разработки нового продукта. Например, в Alias, компании из Торонто (теперь являющейся частью компании Autodesk), которая специализируется на создании трехмерных графических программ, пользовательский опыт максимально учитывается при создании каждого нового продукта. С этой целью в коллективы, занятые созданием новых программ, обычно назначаются по два интерактивных дизайнера (interaction designer), на которых возлагается полная ответственность за учет пользовательского опыта и удобства использования (usability) будущей программы.²⁰ Большое количество полезной информации приходит через обратную связь с пользователями (разного рода опросы, деловые встречи, анкетирования и проверки удобства использования). При этом ожидается, что объединит эту информацию интерактивный дизайнер, отвечающий за поведение программы и обладающий глубоким пониманием задач, решать которые (наилучшим образом) призвано создаваемое программное обеспечение (см. главу 8).

От команды руководителей ждут выработки общего направления, придерживаться которого в процессе создания нового продукта смог бы коллектив разработчиков.

Совместное руководство

Модель минивэна Chrysler NS (Caravan/Voyager/Town and Country) была представлена в 1996 году и стала “хитом” на рынке. Эта модель (в которой была пересмотрена кон-

¹⁹ См. Taylor S., Schroeder K. *Inside Intuit*, Harvard Business School Press, 2003.

²⁰ Miller L, *Case Study of Customer Input for a Successful Product*, Experience Report, *Agile 2005*.

цепция минивэна — добавлена дверь для пассажира рядом с водителем) получила премию автомобиля года (Motor Trend's Car of the Year). Мы (авторы) владели минивэном Chrysler (Town and Country) минивэна выпуска 1997 года в течение восьми лет. Этот автомобиль был создан не главным инженером проекта, а командой экспертов, сумевших выработать общее понимание, чего хотят потребители. Этот коллектив провел глубокое исследование рынка, выполнил подробный анализ технологии развертывания функций (quality function deployment — QFD) и даже осуществил серию экспериментов по простоте и удобству использования, прежде чем принять специфические решения. Эта целеустремленная команда разработчиков проводила вместе много времени (не менее одного дня в неделю), оценивая состояние дел, учась друг у друга и решая текущие проблемы.²¹

Хотя не ясно, кто при этом несет за все ответственность, совместное руководство может явиться удачным подходом для проектов (умеренного объема) по разработке программного обеспечения. Удачным примером использования такого подхода в сообществе Open Source является веб-сервер Apache.²²

Кто за все отвечает

Хорошее маркетинговое и техническое руководство не исключает существования заинтересованного коллектива, который совместно изучает нужды потребителей, рассматривает с разных точек зрения возникающие вопросы и вырабатывает общий взгляд на потребительскую ценность (customer value). Мэри приобрела опыт разработки продукта на 3М²³, где следовали традиционному представлению, что каждый новый продукт нуждается в “чемпионе”. На роль такого чемпиона обычно приходит технический специалист, обнаруживший неудовлетворенный спрос на рынке и создавший технологию для выполнения соответствующей задачи. Однако чемпион не будет успешным, если он не работает с многофункциональным коллективом (фактически создаст предприятие мелкого бизнеса для коммерциализации нового продукта). Подобный коллектив совместно определяет запросы рынка и основные характеристики будущего продукта. Такое взаимодействие чемпиона и коллектива, члены которого представляют все важные функции бизнеса, шло на пользу компании 3М десятилетиями, обеспечивая невероятный рост через создание множества новых продуктов ежегодно. На 3М общие ключевые решения обычно принимал коллектив разработчиков, однако в конце концов ответственность за разработку продукта и преодоление всех проблем нес именно чемпион. Аналогично, в компании Toyota основным принципом является: “Коллективный труд — ключ к выполнению высококачественной работы, но ответственность должен нести кто-то один”.²⁴ В своей книге *Who Has the D? How Clear Decision Roles Enhance Organizational Performance* (“Кто принимает решения? Как ясность в том, кто принимает решения, может повысить эффективность организации”) авторы Пол Роджерс (Paul Rogers) и Марция Бленко (Marcia Blenko) утверждают, что хорошо отработанный механизм принятия решений яв-

²¹ Sobek, Там же. Автор лично наблюдал работу данного коллектива и рассказал об этом в своих тезисах.

²² Fielding R.T. *Shared Leadership in the Apache Project*, Communications of the ACM, April 1999.

²³ Предприятие по производству видеокассет.

²⁴ Morgan J., Liker J. *The Toyota Product Development System: Integrating People, Process, and Technology*, Productivity Press, 2006, p. 103.

ляется отличительной чертой успешных организаций, и что ясность в том, кто какие решения принимает, является залогом принятия хороших, своевременных решений.²⁵

Назначение “принимающего решения”

Я руководила созданием нескольких новых продуктов в компании 3M. По моему мнению, это очень ответственная, но и благодарная роль. На еженедельных собраниях коллектива, занятого разработкой соответствующего продукта, естественно, нередко возникали разногласия, поскольку представители разных специальностей имели различные точки зрения. Подчас не было ясно, собирается ли ответственный за качество когда-либо выпустить продукт на рынок, и понимает ли специалист по маркетингу, что процесс разработки продукта — это тяжкий труд. Я заметила, что разногласия обычно касаются не достоинств и недостатков решения, а того, кому именно следует решать.

Поэтому я сочла полезным каждый раз назначать специальное лицо, ответственное за принятие данного решения. При этом я выбирала специалиста, наиболее подходящего для текущего случая. Ответственный за качество одобрит выпуск окончательной версии, маркетолог решит, какими свойствами должен обладать продукт, разработчик скажет, возможно ли реализовать ту или иную функцию, и т.д.

После назначения принимающего решения тон дискуссии немедленно изменился. Члены коллектива начали пытаться воздействовать на принимающего решения с целью убедить его стать на их точку зрения. При этом ответственный за принятие решения постарался сформировать собственную точку зрения, учитывающую мнения всех сторон. После этого все последующие дискуссии обычно касались достоинств и недостатков того или иного решения, а не того, кому следует решать. (Мэри Поппендик)

Многофункциональные коллективы

Совершенные продукты создаются многофункциональными командами. Когда руководство компании Intuit приняло решение о создании программы Quicken Rental Property Manager, была создан коллектив, включавший всех специалистов, необходимых для создания продукта и продвижения его на рынке, а “не только разработчиков программного обеспечения”.²⁶ Членам нового коллектива приходилось работать вместе и раньше — у них за плечами был опыт 20-летнего совершенствования программы Quicken. Однако новый продукт с маркой Quicken предстояло создать впервые за много лет. Задача коллектива состояла в том, чтобы решить соответствующую проблему потребителей — ни больше, ни меньше, — создав программу с минимальным набором функций и в кратчайшие сроки.

Коллектив разработчиков пришел к выводу, что его основная задача состоит в том, чтобы учиться — узнать все о задачах, стоящих перед будущими пользователями программы с тем, чтобы эта программа включала как раз необходимый набор функциональных возможностей. Коллектив регулярно отчитывался перед руководством компании о состоянии дел, получал общие указания, а затем работа продолжалась до следующей “контрольной точки”. На каждой из таких встреч коллектив разработчиков рассказывал, что нового было выяснено о работе будущих пользователей программы и что планируется выяснить до следующей “контрольной точки”. За исключением общих указаний, получаемых на таких встречах, решения по будущему продукту принимались в коллективе, создающем программу.

²⁵ Rogers P., Blenko M. *Who Has the D? How Clear Decision Roles Enhance Organizational Performance*, Harvard Business Review, January 2006.

²⁶ Сообщение Soni Meckem на конференции Lean Design & Development 2005, Chicago, March, 2005.

Команда создателей Quicken Rental Property Manager начала с того, что отправилась опрашивать потенциальных пользователей. При этом было замечено, что вопросы разработчиков отличаются от вопросов маркетологов. В ходе опроса, благодаря различным точкам зрения, все члены коллектива получили более глубокие представления о том, каким следует быть будущему продукту. Выяснилось также, что при таком подходе процесс разработки не требует так много бумажной работы, как это представлялось ранее, поскольку все члены коллектива слышали от пользователей одно и то же.

После этого члены коллектива полностью погрузились в процесс создания нового продукта. Через год программа Quicken Rental Property Manager была выпущена для оценки пользователей, которые были в восторге от ее понятного интерфейса, незамысловатого процесса установки и простоты использования.

Технологичность

Один из наибольших эффектов в процессе производства имеет место, когда производственные инженеры приглашаются в коллектив разработчиков будущего продукта. При этом каждое конструкторское решение, прежде чем быть одобренным, оценивается с точки зрения простоты производства. В результате появляются продукты, которые не только легко и быстро производить, но и просто обслуживать. Это называется технологичностью конструкции (design for manufacturability).

Технологичность изделия

Давным-давно на нашем предприятии видеокассеты собирались вручную. Руководитель предприятия предлагал руководителям всех отделов регулярно тратить некоторое время, работая на сборочной линии, поэтому я время от времени работала в ночную смену, собирая видеокассеты. Там был один винт, доступ к которому был затруднен для обычной отвертки, и поэтому мы использовали для него специально изготовленную согнутую отвертку. Каждый раз, когда мне приходилось тянуться за этим специальным инструментом, а затем мучиться, вкручивая винт, я мечтала о том, чтобы привести конструкторов на сборочную линию и заставить их собрать несколько кассет.

Позже, когда конструкторы стали подчиняться мне, я всегда требовала, чтобы их разработки просматривались представителями производства, прежде чем быть утвержденными. Сначала конструкторы были поставлены в тупик, но очень скоро почувствовали уважение к реалиям производства. Вскоре чуть ли не через каждые два дня они сами несли свои чертежи к контролеру производства, надеясь получить полезные замечания. Не стоит и говорить, что новые изделия было легко собирать. (Мэри Поппендик)

В разработке программного обеспечения эквивалентом технологичности является функциональность. Каждый коллектив разработчиков должен иметь в своем составе специалистов, которые бы побуждали коллектив учитывать, какие проблемы могут возникнуть, когда программное обеспечение будет использоваться в реальных условиях, и что нужно сделать, чтобы избежать их возникновения.

Все, что может случиться, рано или поздно случится

Коллега, бывший разработчик, который теперь производственный менеджер, имел любимую поговорку, которую разработчикам полезно запомнить: “Все, что может случиться на производстве, рано или поздно случится, и необходимо заранее предусмотреть меры, позволяющие выявить и локализовать проблему, а затем устранить ее последствия”. Недавно он писал, рассказывая о своей последней проблеме: “Компания “XYZ” практически еженедельно выпускала новые версии программного обеспечения. Эти версии как правило были вполне работоспособными и не демонстрировали слишком много регрессивных ошибок. Основная проблема состояла в том, что ПО

проявляло нестабильность либо с течением времени, либо с увеличением нагрузки. При прохождении функциональных тестов на качество оно (ПО) часто проявляло несоответствие требованиям производства.”

Я предложила проанализировать источник каждого отказа. Оказалось, что это уже делалось.

“Исчерпывающий анализ неизменно является частью реакции на каждую проблему. В результате была обнаружена не единственная ошибка, а целый набор ошибок, которые все можно в конечном счете отнести на счет небезопасной практики программирования.”

В этом примере коллектив разработчиков не ощущал ответственности за созданное программное обеспечение (после передачи его заказчику) и был не заинтересован в результатах анализа отказов. (Мэри Поппендик)

Разработка на заказ

То обстоятельство, что программное обеспечение разрабатывается только для одного заказчика, не освобождает соответствующую компанию от обязанности создать потребительскую ценность и порадовать потребителя. По-прежнему, целью является создание программного обеспечения, которое явится ценностью для организации, платящей за него. В действительности потребность в хорошем руководстве и многофункциональной команде, вероятно, только возрастает для продуктов, создаваемых на заказ, по сравнению с продуктами, которые затем предлагаются на рынке. Дело в том, что отсутствие конкуренции может ослабить ориентирование на нужды потребителя, которое является отличительной чертой успешных компаний, занятых созданием программного обеспечения.

От проектов к продуктам

Программное обеспечение на заказ часто создается как проект, однако мы полагаем, что разработка его в качестве продукта представляет собой более полезную модель. Одно из интересных обстоятельств, связанных с проектами, состоит в том, что они (проекты) имеют тенденцию быть профинансированными вначале и сразу (рис. 3.2). После выделения средств естественно, возникает вопрос: *что и когда мы получим в обмен на наше инвестирование?* Ответы на подобные вопросы рассматриваются как обязательства (в конце концов, деньги-то были заплачены!). Иными словами, успех проекта определяется на основе того, не вышел ли проект за пределы выделенной суммы, соблюден ли график и обладает ли созданное программное обеспечение (если цель проекта — создание ПО) всеми необходимыми характеристиками.



Рис. 3.2. Типичный график финансирования проекта

С другой стороны, продукты, как правило, финансируются поэтапно (рис. 3.3). Поэтапное финансирование — это четкий сигнал для всех, что потребители ожидают от создаваемого продукта на каждом этапе все большего соответствия его требованиям. Успех продукта обычно измеряется на основе занятой им доли рынка и прибыльности, которой он в конце концов достигает.

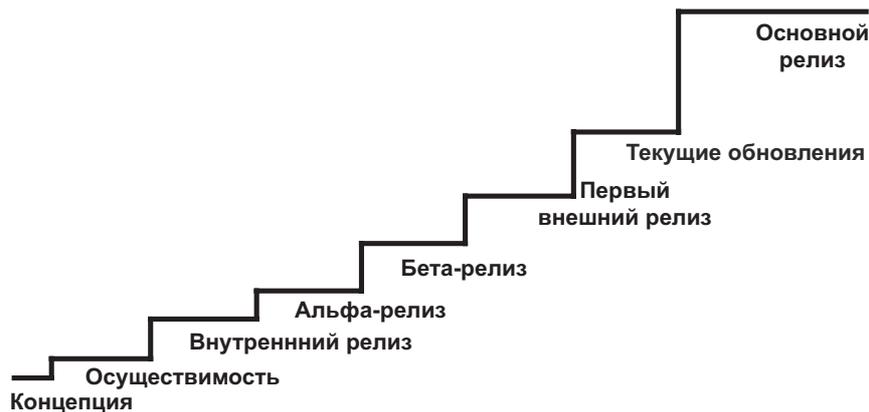


Рис. 3.3. Типичный график финансирования продукта

Имеются и иные отличия между проектами и продуктами. Проекты имеют начало и (очевидно) окончание. Продукты, с другой стороны, имеют начало и (как надеются разработчики) не имеют конца. Программное обеспечение больше похоже на продукт, чем проект, поскольку хорошие программы используются и подвергаются усовершенствованиям в течение долгого времени. Как отмечалось в главе 2, большая часть всего программного обеспечения разрабатывается после выпуска первой версии.

Созданную на заказ систему нельзя закончить

Когда мы приступали к внедрению системы управления технологическим процессом на предприятии, нас предупредили, что будет множество запросов на изменения. Это показалось хорошим знаком. Если наша система была нужна предприятию, следовало ожидать множества идей, как лучше использовать технологию.

Поскольку изначально не предполагается, что управляющую систему можно “закончить”, мы всегда создавали наши системы с возможностью изменения конфигурации. Подобные системы обычно создаются для управления таким оборудованием, как насосы, вальцы, печи и т.п., поэтому некоторые оригинальные идеи часто поступают от операторов, работающих на подобном оборудовании. Так как линия по изготовлению ленты постоянно использовалась для производства различных продуктов, управляющая система должна была позволять оператору очень легко изменять используемые скорость, давление и температуру.

Даже при наличии системы с возможностью изменения конфигурации мы любили уходить из цеха с длинным списком изменений и дополнительных функциональных возможностей, которых хотели операторы линии. Для нас это была высшая оценка. (Мэри Поппендик)

Для выполнения каждого проекта, как правило, создается новый коллектив. С другой стороны, продукты обычно разрабатываются постоянными коллективами, которые продолжают поддерживать их и в дальнейшем. Поддержку программного обеспечения лучше

осуществит постоянный коллектив, поскольку знание запросов заказчика, программный код и т.п. трудно передать другим разработчикам как “наследство”. А большинство программного обеспечения обычно используется долго, подвергаясь множеству усовершенствований и улучшений.

Сотрудничество между ОИТ и бизнесом

В отчете *When IT's Customers Are External*²⁷, опубликованном в онлайн-журнале *McKinsey Quarterly* Симон Мак-Гиббон (Simon MacGibbon) и его соавторы выражают мысль, что подразделения информационных технологий (ОИТ) крупных корпораций, которые непосредственно взаимодействуют с бизнесом (в котором участвует данная корпорация), должны управляться и быть организованы как компании, производящие программное обеспечение. В отчете указывается, что компании-производители программного обеспечения имеют три отличия от внутренних (для крупных корпораций) ОИТ.

1. Компании-производители программного обеспечения проводят исследования рынка, чтобы по-настоящему понять запросы потребителей с тем, чтобы создавать продукты, которые будут продаваться. Если это будет сделано плохо, соответствующие компании-производители не продержатся долго. Собственно, подобные исследования выполняются регулярно в поиске наилучших ответов на запросы рынка. Поскольку такие компании не могут претендовать на то, чтобы их потребители участвовали в определении проектных решений для будущего продукта, они придумывают разного рода консалтинговые службы, которые призваны узнавать из первых рук, что именно нужно пользователям. Что касается ОИТ, то они часто пренебрегают маркетинговыми исследованиями, предполагая, что это уже сделал кто-то другой, и удивляются, когда созданный ими продукт не отвечает потребностям бизнеса.
2. Компании-производители программного обеспечения действуют на рынке, где имеет место жесткая конкуренция, поэтому они вынуждены поддерживать свои цены на достаточно низком уровне. И они создают достаточно простые продукты, разработка и последующая поддержка которых рентабельны, в то же время заботясь, чтобы предлагаемый набор функциональных возможностей привлекал пользователей и побуждал их покупать их программное обеспечение. В то же время ОИТ часто исходят из того, что перечень функциональных возможностей будущего программного обеспечения должен включать все, в чем может нуждаться бизнес, в том числе и очень дорогие функциональные возможности. У этих отделов мало стимулов поддерживать баланс между затратами и эффектом, что вынуждены делать компании-разработчики, чтобы оставаться на плаву.
3. Компании-производители программного обеспечения понимают, что если пользователи их продуктов не будут успешны в своем бизнесе, они (компании-производители) не будут иметь жизнеспособного бизнеса, поэтому производители пользуются каждой возможностью помочь компаниям-пользователям добиться успеха. В то же время ОИТ очень часто полагают, что успех в использовании созданных ими систем полностью зависит от подразделений корпорации, занимающихся

²⁷ MacGibbon S. P., Schumacher J. R., Tinaikar R. S. *When IT's Customers Are External*, McKinsey Quarterly, Q1 2006.

соответствующим бизнесом. Хотя и верно, что эти корпоративные подразделения должны нести ответственность за успех использования соответствующего программного обеспечения, так же верно, что усилия ОИТ нельзя признать успешными, если созданные им продукты не вносят соответствующий вклад в успех бизнеса.

Многие ОИТ для разработки своего программного обеспечения используют модель проекта. А по нашему мнению, для создания здоровых отношений между таким отделом и соответствующим бизнесом больше подходит модель продукта, поскольку стимулы, связанные с этой моделью, с большей вероятностью смогут создать отношения сотрудничества. Так как ОИТ является частью корпорации, не существует реальных причин для возникновения отношений “мы и они”, на которые ориентированы проекты. В этом случае разработчикам нет необходимости делать то, что важно для потребителей программного обеспечения. А чтобы предоставить потребительскую ценность, наиболее необходимую для бизнеса, в кратчайшие сроки и за минимальную цену, помочь в эффективном использовании системы, а затем участвовать в совершенствовании программного продукта, необходимо сотрудничать со своим бизнес-партнером.

Путь к успеху

Однажды на конференции я разговорилась со старшим представителем информационной службы крупной финансовой холдинговой компании. По образованию он был инженером-электриком, которому однажды было поручено “навести порядок” в ОИТ.

“Это было более трех лет назад, и тогда я потерпел неудачу. Я перепробовал несколько методов и спустя два года потерял надежду и вернулся к прежней работе.

“Однако еще через год меня попросили попробовать снова и на этот раз я поступил иначе. Я разделил отдел на две группы: отвечающую за эксплуатацию и занимающуюся разработкой. Затем я разделил группу разработки на шесть небольших коллективов. Каждый коллектив должен был продавать созданное им ПО представителям бизнеса и результаты их деятельности оценивались по полученной от использования этого ПО прибыли в сравнении с его стоимостью.

“Прошло только шесть месяцев, но уже видно, что принятые меры оказались чрезвычайно успешными. Не понимаю, почему это не пришло мне в голову раньше.” (Мэри Поппендик)

Ответственность

В крупных корпорациях традиционно имеются отделы информационных технологий (ОИТ), часто независимые от бизнеса, который они обслуживают. Это объясняется тем, что корпорации обычно стремятся иметь стандартную информационную инфраструктуру, а также тем, что технические разработки легче осуществлять, когда соответствующие специалисты являются частью организации. Однако при таком положении дел имеет место неясность, кто является ответственным за результаты деятельности, связанной с разработкой программного обеспечения, а также как оценивать эти результаты.

Вопрос об ответственности не нов для ОИТ. Он возникает каждый раз, когда члены одного коллектива разработчиков выполняют работу для разных подразделений корпорации с различными способами оценки продуктивности. Например, управление разработкой встроенного программного обеспечения может осуществляться отдельно от разработки аппаратных средств.

Не особенно эффективным подходом является распределение ответственности по всем этапам процесса разработки, когда одной части коллектива разработчиков предлагается предъявлять “требования” другой части. Подобный подход имеет тенденцию де-

лать неясной конечную цель деловой активности совместного предприятия и часто ведет к посредственным результатам. Гораздо эффективнее, когда члены многофункционального коллектива делят ответственность за достижение результатов, которые бы оправдывали финансирование их работы.

Однако в конце должна быть единая ответственность по общим результатам инвестирования в ОИТ. Мы полагаем, что эта ответственность должна быть перед бизнесом, который осуществляет финансирование. Когда руководители бизнеса будут относиться к инвестициям в ОИТ с той же строгостью, с какой они управляют своим бизнесом, эти инвестиции наверняка дадут значительные результаты.²⁸

Попробуйте это

1. Используя модель Кано, выполните анализ процесса разработки, в котором вы участвуете. Для этого составьте список функциональных возможностей — на высоком уровне, — а затем перенесите пункты этого списка на индексные карточки. На большом листе бумаги создайте модель Кано и прикрепите его к стене. Далее прикрепите карточки к соответствующим областям модели Кано. Как много функциональных возможностей попало в категорию “Восхищение и удивление”? Сколько это процентов?
2. В главе 10 мы описываем показатель, являющийся хорошей мерой удовлетворенности пользователя, который известен как “чистая оценка готовности заказчика рекомендовать организацию, создавшую данный продукт” (net promoter score). Определите оценку готовности рекомендовать вашу организацию для ваших пользователей. Для этого проведите простой опрос, чтобы выяснить у пользователей, насколько вероятно, что они рекомендуют ваш продукт или услугу коллеге. Предложите им оценить это по 10-балльной шкале. Какой процент ответов составляют оценки 9 и 10? Этот процент определяет число “промоутеров” вашей организации. Какой процент ответов составляют оценки от 1 до 6? Этот процент определяет число “очернителей” вашей организации. Вычтите число процентов “очернителей” из числа процентов “промоутеров”. Полученный результат положительен или отрицателен?
3. Составьте список программ или проектов в вашей организации, которые в настоящее время открыты. Для каждого пункта списка укажите имя имеющегося в данный момент руководителя (или руководителей). Как много руководителей имеет каждая программа или проект? Есть ли среди них кто-нибудь, способный осуществлять одновременно маркетинговое и техническое руководство? Существует ли какая-либо связь между состоянием проекта и качеством руководства (или отсутствием такового)?
4. Многофункциональные коллективы: имеете ли вы производственных инженеров в вашем коллективе? Проводятся ли регулярные консультации с кем-нибудь из группы технической поддержки при разработке новых функциональных возможностей? Участвуют ли тестировщики в процессе разработки с самого начала? На каком этапе привлекаются создатели инструкций (technical

²⁸ Lohmeyer D., Pogreb S., Robinson S. *Who's Accountable for IT?*, McKinsey Quarterly, December 7, 2004.

writers)? Относятся ли у вас к представителям заказчика как к полноправным членам вашего коллектива?

5. Проекты/Продукты: финансирование разработки у вас осуществляется поэтапно или сразу? Что является критерием успеха разработки? Сохраняются ли коллективы после завершения разработки или люди обычно переводятся в другие коллективы? Сохраняет ли коллектив созданный им программный код? Кто несет ответственность за прибыль от инвестиций в разработку программного обеспечения?