ГЛАВА 3

API как продукт

> Все, что создают люди — будь то продукт, коммуникация или система, — результат воплощения вдохновения в жизнь.
>
> *Мэгги Макнаб*

Фразу «API как продукт» (AaaP) часто упоминают в разговоре о компаниях, создавших и поддерживающих успешные программы API. Это перефразированное выражение «...как услуга», которое часто используется в технических кругах («ПО как услуга», «платформа как услуга» и т. д.). Как правило, оно отображает важную роль разработки, применения и реализации API, ведь API — это *продукт*, полностью заслуживающий продуманного дизайна, создания прототипов, исследования клиентской базы, тестирования, длительного контроля и технической поддержки. Обычно эта фраза означает: «Мы относимся к нашим API так же, как к любым другим нашим продуктам».

В этой главе мы рассмотрим подход AaaP и то, как его использовать для лучшего проектирования, развертывания ваших API и управления ими. Этот подход включает в себя осознание того, какие решения важны для успеха ваших API и в каком отделе их должны принимать (см. главу 2). Он поможет вам обдумать, какую работу надо централизовать, а какую можно успешно децентрализовать, где лучше задействовать ограничения и поощрения и как измерить степень влияния этих решений для быстрой адаптации ваших продуктов (API).

При создании новой продукции для клиентов приходится принимать множество решений. Неважно, разрабатываете ли вы плееры, ноутбуки или API для очереди сообщений. Во всех этих случаях вам нужно: 1) знать свою целевую аудиторию, 2) понимать и решать ее первостепенные проблемы и 3) обращать

внимание на предложения клиентов об улучшении продукции. Эти важнейшие задачи можно сформулировать в виде трех ключевых уроков, на которых мы сосредоточимся в этой главе:

- дизайн-мышление как способ убедиться в том, что вы знаете свою целевую аудиторию и понимаете ее проблемы;
- поддержка новых клиентов как способ быстро показать им, как успешно пользоваться вашим продуктом;
- опыт разработчика как способ управления жизненным циклом продукта после его выпуска и для выработки идей будущих модификаций.

Вскоре мы узнаем о силе дизайн-мышления и клиентской поддержки на примере таких компаний, как Apple. Увидим, как Джефф Безос помог подразделению Amazon Web Services (AWS) создать устав реализации, обеспечивающий получение четкого и предсказуемого опыта разработчика. Мы общались с представителями многих компаний. Большинство из них понимают, что такое AaaP, но не могут применить этот подход на практике. Но во всех организациях, на счету которых немало успешных продуктов API, поняли, как эффективно использовать три упомянутых ключевых урока, первый из которых связан с тем, как ваши команды *обдумывают* то, что создают.

Программируемая экономика, основанная на API

API — это интерфейсы для создания программируемой экономики. Но они должны быть разработаны так, чтобы их можно было найти, масштабировать и они выполняли заявленные функции для решения проблем разработчиков. Для этого компаниям нужно правильно подходить к управлению ожиданиями и стремлениями своих сообществ разработчиков. Именно здесь в игру вступают отношения с разработчиками. Они устанавливают связь между тем, что может предоставить каждый из ваших API, и квалифицированными специалистами, которые будут внедрять их в другие приложения, — разработчиками. Чуть позже мы рассмотрим, как API меняют правила игры в программируемой экономике, обеспечивая больший охват, масштабируемость и повсеместность, и рассмотрим роль отношений с разработчиками в контексте управления API, пропаганды и евангелизации.

Чтобы говорить о значительности API, нужно понять, почему они так важны для бизнес-стратегии. В 2011 году один из самых известных инвесторов Кремниевой

долины и основатель Netscape Марк Андриссен заметил: «Программное обеспечение поглощает мир».

До этого возможности информационных технологий (IT) были интегрированы в организации для поддержки бизнеса. Но с появлением сетевых и инфраструктурных технологий, позволивших реализовать модель «как услуга», когда ПО можно запускать из чужой инфраструктуры (SaaS, PaaS, Iaas), IT *стали* бизнесом, где третьи стороны могли предлагать самообслуживание, автоматизированные, программируемые и отслеживаемые функции, позволяя бизнесу сохранять полный контроль. А что стало ключом, позволившим этим сторонним организациям предоставлять возможности *«как услуга»*? Это открытые API.

Эти программируемые интерфейсы позволяют компаниям и их приложениям открывать свой бизнес для третьих сторон и выходить за пределы собственных возможностей развития. Вне организации находится больше разработчиков, ресурсов, идей, знаний рынка, энтузиазма, инноваций и капитала, чем внутри. Так почему бы не открыть активы и возможности для большего числа экономических и общественных субъектов, которые могут быть вовлечены в создание ценности?

Раньше конкуренция шла по принципу «продукт против продукта», но теперь он сменился на принцип «платформа против платформы», а затем перерос в принцип «экосистемы против экосистем». API — это программируемые интерфейсы, которые ведут от одной формы к другой.

Цена, продвижение, продукт, место → везде

Классические менеджеры знают 4P от гуру маркетинга Майкла Портера: Price, Promotion, Product, Place (цена, продвижение, продукт, место). Это четыре переменные, которыми вы управляете в маркетинговой стратегии продукта.

Но в цифровом мире, где API «съедают» ПО, по словам Эндрю Босворта, главы отдела развития Facebook, «побеждает не лучший продукт, а тот, который используют все»[1]. Это относится и к вашим API. Иногда опыт использования поставляемого продукта может иметь большее значение для клиента по сравнению с некоторыми дополнительными функциями, которые вы захотите добавить. Итак, опыт разработчика, который вы предоставляете пользователям своего API, будет иметь конкурентное преимущество.

В цифровом мире, где IT-возможности предоставляются как услуга через API, цель не в том, чтобы быть в нужном месте, а в том, чтобы быть везде. Речь идет не о том месте, которое вы можете контролировать, а обо всех возможных местах,

[1] *Balfour B.* Growth Wins («Рост побеждает») // Reforge (запись в блоге), последнее изменение от 25 июля 2018. https://oreil.ly/UJDIU.

в любых приложениях. Например, банковская и финансовая индустрия претерпевает изменения из-за API, позволяющих внедрять встроенные финансы.

Пол Рохан, автор и эксперт по Open Banking API, объясняет, что будущее банковского дела не «в банке», а везде, где нужно банковское обслуживание: встраивание в сторонние клиентские сервисы, на платформах недвижимости, в приложениях для планирования свадеб, в виджетах на сайтах автодилеров, на сайтах электронной коммерции — везде[1]. Благодаря API банки могут иметь доступ к любой клиентской базе, которой они не владеют, получая при этом возможность предоставлять выгодные предложения.

В сообщении блога, опубликованном в 2020 году, идейный вдохновитель платформы и отраслевой консультант Саймон Торранс подсчитал, что в течение пяти лет возможности встроенного финансирования составят 7,2 миллиарда долларов. Это вдвое превышает общую рыночную стоимость нынешних банковских и финансовых услуг[2]. Когда вы повсюду, вы получаете новый, более широкий охват.

Давайте посмотрим, какие изменения произошли в этой области за последние 20 лет. Для распространения своего продукта в цифровом формате в 2000 году вам нужен был веб-сайт. В 2010 году для этой цели вам понадобилось мобильное приложение. В 2020-м — API. Реалии таковы, что новый способ завоевания цифрового пространства — это интеграция в сторонние сайты и приложения. Речь идет уже не о контроле над одним-двумя каналами, а о внедрении в максимально возможное число тех каналов, где находятся ваши пользователи.

Как описывает Крис Андерсон в своей книге «Длинный хвост» (Hachette), если первые каналы распространения (веб- и мобильные) составляли значительную часть общего трафика, то «длинный хвост» остальных самых маленьких ниш и каналов фактически представлял собой растущий со временем трафик, который иногда становился даже больше, чем традиционные каналы.

Некоторые приложения, такие как Salesforce или eBay, получают основную часть своего трафика через сторонние платформы, а не через собственный сайт или мобильные приложения. Более 50 % их трафика поступает через API. Единственная проблема для компаний была в том, что слишком сложно и дорого появляться во всех этих каналах одновременно. Но теперь можно обращаться к нескольким каналам с помощью одних и тех же API.

---

[1] *Rohan P.* Driving Business Growth and Brand Strategy in the Api-Powered Age of Assistance («Стимулирование роста бизнеса и стратегия бренда в эпоху помощи, основанной на API») // APIdays, Лондон, 2019. https://oreil.ly/IcxbV.

[2] *Torrance S.* Embedded Finance: A Game-Changing Opportunity For Incumbents («Встроенные финансы: возможность изменить правила игры для сотрудников»). 10 августа 2020 г. https://oreil.ly/L6I3n.

Теперь API — продукт нашего программируемого мира. В следующей главе вы научитесь обращаться с ними как с продуктами, начиная с ознакомления с проектом и начальных шагов и заканчивая опытом разработчиков и успешными интеграциями.

Дизайн-мышление

Среди разработчиков продукта компания Apple известна, в частности, благодаря своей способности задействовать *дизайн-мышление*. Например, описывая работу по созданию Apple Mac OS X, один из ключевых разработчиков ПО Корделл Ратклафф сказал: «Мы сосредоточились на том, что, по нашему мнению, было нужно людям, и на том, как они взаимодействуют с компьютером»[1].

Этот подход затем отразился на практике. «Для создания идеи продукта нужны три отчета: о требованиях маркетинга, о технических требованиях и об опыте пользователей», — однажды объяснил бывший вице-президент Apple и один из первопроходцев в области разработки взаимодействия человека и компьютера Дональд Норман[2].

Такое внимание к потребностям пользователей привело Apple к созданию стабильного бизнеса. Непрерывная цепочка продуктов, выпускаемых в течение нескольких десятилетий, создала им репутацию компании, определяющей новые тенденции в технологиях, и помогла неоднократно занимать большой сегмент рынка.

Тим Браун, генеральный директор IDEO, расположенной в Калифорнии фирмы по разработке и консультированию, определяет термин *«дизайн-мышление»* так[3]:

> Раздел дизайна, где используются чутье и методы разработчика, чтобы продукт соответствовал требованиям пользователей и был технологически осуществимым, а конкурентная бизнес-стратегия могла извлечь из него потребительскую ценность и возможность для рынка.

---

[1] *Meade S.* Steve Jobs: Mac OS, Designed by a Bunch of Amateurs («Стив Джобс: Mac OS, разработанная кучкой дилетантов») // Synap Software, LLC (блог), 16 июня 2007 г. https://oreil.ly/jRURa.

[2] *Turner D.* The Secret of Apple Design («Секрет дизайна Apple») // MIT Technology Review, 1 мая 2007 г. https://oreil.ly/ehrgv.

[3] *Brown T.* Design Thinking («Дизайн-мышление») // Harvard Business Review, июнь 2008 г. https://oreil.ly/VRA7Y.

Здесь важно пояснить несколько пунктов. Для наших целей нам нужно сосредоточиться на понятиях «соответствие требованиям пользователей» и «конкурентная бизнес-стратегия».

Соответствие требованиям пользователей

Одна из ключевых целей разработки API — решение проблем. Поиск проблем, требующих решения, и определение их приоритетности — часть задачи подхода AaaP. Это ответ на вопрос «что делать?» в области API. Еще более важная составляющая — ответ на вопрос «для кого?». Кому вы облегчите жизнь с помощью этого API? Правильное определение целевой аудитории (ЦА) и ее проблем — первый шаг к тому, чтобы убедиться, что вы создаете нужный продукт: он эффективно работает и часто используется ЦА.

Клейтон Кристенсен из Гарвардской бизнес-школы называет процесс осознания потребностей вашей ЦА теорией «работы, которые нужно выполнить». Он говорит: «Люди не просто покупают товары или услуги, они “нанимают” их, чтобы продвинуться в определенных обстоятельствах»[1]. Люди (потребители) хотят продвинуться (решить проблемы), и они будут применять (или арендовать) те продукты и услуги, которые, по их мнению, помогут им в этом.

МОЖНО ЛИ ПРИМЕНЯТЬ AAAP КАК К ВНУТРЕННИМ, ТАК И К ВНЕШНИМ API?

Да. Но, возможно, с разными затратами времени и ресурсов — это мы обсудим в следующем разделе. Это один из тех уроков, которым Джефф Безос поделился в «Уставе Безоса», позволившем Amazon открыть изначально внутреннюю платформу AWS для использования в качестве приносящего доход внешнего API. Поскольку Amazon с самого начала применяла подход AaaP, предложить внешним пользователям внутренний API было не только *возможно* (то есть безопасно), но и *выгодно*.

В большинстве компаний IT-отдел помогает решать проблемы. Обычно их клиенты — сотрудники той же компании (внутренние разработчики). Иногда это важные бизнес-партнеры или даже анонимные разработчики сторонних приложений (внешние разработчики).

Каждая из этих групп разработчиков (внутренняя, партнерская и внешняя) сталкивается со своим набором проблем и по-своему обдумывает и решает их.

[1] Jobs to Be Done («Работы, которые нужно выполнить») // Институт Кристиансена, последнее изменение от 13 октября 2017 г. https://oreil.ly/l1s63.

Дизайн-мышление побуждает команды лучше узнать свою аудиторию, прежде чем приступать к созданию API в качестве решения. Мы исследуем эту тему в разделе «Познакомьтесь с целевой аудиторией» на с. 87.

Конкурентная бизнес-стратегия

Другая важная часть планирования дизайна — определение *конкурентной бизнес-стратегии* вашего API. Бессмысленно тратить много денег и времени на продукт, который не окупится. Даже когда вы стараетесь разработать правильный продукт для правильной ЦА, важно убедиться, что ресурсы потрачены не впустую и у вас есть четкое представление о доходе после запуска API.

У большинства компаний ограничено количество ресурсов, которое можно выделить на создание API для решения проблем. Поэтому огромное значение приобретает выборка этих проблем. Иногда мы сталкиваемся с компаниями, где API не решают важных бизнес-задач, разбираясь вместо этого с хорошо знакомыми проблемами IT-отдела: открытием таблиц данных или автоматизацией внутренних процессов отдела. Все это, конечно, важно, но такие решения не сильно влияют на ежедневные бизнес-операции и не приближают компанию к достижению ежегодных целей в сфере продаж или производства.

Непросто определить значимые для вашего бизнеса проблемы. Руководству может быть сложно сообщить о целях компании так, чтобы IT-отдел мог легко их понять. И даже когда IT-команда понимает, какие проблемы важны для компании, у нее может не оказаться подходящих методик измерения для подтверждения их предположений. Поэтому важно иметь стандартизированный способ сообщения ключевых бизнес-целей и соответствующих показателей эффективности. Этот аспект пути к успеху вашего API мы обсудим подробнее в разделе «Измерения и ключевые этапы» на с. 193.

Устав Безоса

Запускать успешную программу с API, способную преобразить вашу компанию, непросто независимо от возраста организации. Одна из самых уважаемых корпораций, проработавших этот процесс и не прекращающих преобразования даже десять лет спустя, — это Amazon, создавшая платформу AWS.

Разработанная в начале 2000-х, эта платформа считается шедевром, виртуозно исполненным изобретательной командой специалистов в области IT и бизнеса. Хотя платформа AWS добилась огромного успеха, она возникла из-за внутренней потребности — большого недовольства тем, что IT-программы Amazon тратили слишком много времени на обработку и доставку запросов

бизнес-команды. Команда AWS работала слишком медленно, и вначале продукт не соответствовал требованиям и на техническом (объем работы), и на деловом (качество продукта) уровнях.

Как рассказывает нынешний генеральный директор AWS Энди Джесси, команда AWS (вместе с генеральным директором Amazon Джеффом Безосом и другими сотрудниками) потратила немало времени на определение того, что у них получается удачно и что нужно для разработки и создания базового набора общедоступных сервисов на межоперационной платформе[1].

На разработку плана ушло более трех лет, но в итоге он лег в основу знаменитой новой платформы, действующей по принципу «инфраструктура как услуга» (infrastructure-as-a-service, IaaS). Бизнес, который принес своим разработчикам 45 млрд долларов (прибыль в размере 13,5 млрд долларов США за февраль 2021 года), был создан только благодаря вниманию к деталям и неустанным попыткам улучшить изначальную идею. Примерно так же, как Apple изменила представление покупателей о портативных устройствах, AWS изменила представление бизнеса о серверах и другой инфраструктуре.

AWS смогла изменить мнение об API *внутри* компании во многом благодаря тому, что сейчас известно как *«Устав Безоса»*. Стив Йегги, бывший руководитель разработки ПО в Amazon, описал этот устав в статье «Тирада о Google-платформах»[2]. Один из важнейших пунктов этой записи из блога в том, что Безос выпустил устав, по которому все команды должны раскрывать свой функционал с помощью API и пользоваться функционалом других команд тоже через них. Иными словами, что-то делать можно только через API. Также он потребовал, чтобы все API были созданы и разработаны *так, словно они будут доступны за пределами компании*.

Мысль о том, что «API должны быть внешними», была еще одним важным требованием, повлиявшим на создание, разработку и управление API.

Итак, дизайн-мышление заключается в том, чтобы соответствовать потребностям вашей аудитории и поддерживать жизнеспособные бизнес-стратегии при принятии решения о том, какие API достойны ваших ограниченных ресурсов и внимания. Как это выглядит на практике? Как можно применить знания о продукте к процессу управления API, чтобы он соответствовал подходу «API как продукт»?

---

[1] *Miller R.* How AWS Came to Be («Как возникла AWS») // TechCrunch, 2 июля 2016 г. https://oreil.ly/OtRyN.

[2] Stevey's Google Platforms Rant («Тирада Стива о Google-платформах») // GitHub Gist, 11 октября 2011 г. https://oreil.ly/jxohc.

Применение дизайн-мышления к API

Вы можете поднять статус API с утилит до продуктов, применяя дизайн-мышление в процессе их создания и разработки. Некоторые компании, с представителями которых мы общались, поступают именно так.

Они решили, что их API, в том числе те, что предназначены для внутреннего пользования, заслуживают того же уровня внимания, изучения и разумного дизайна, что и другие их продукты. Для некоторых компаний это значит, что они должны обучить своих разработчиков API и других сотрудников IT-отдела принципам дизайн-мышления. Для других это означает создание внутри организации мостика между группами, занимающимися дизайном продукта и API. В нескольких компаниях, с которыми мы работали, наблюдались сразу оба этих процесса: обучение разработчиков дизайн-мышлению и усиление связи между командой продукта и разработчиками.

Полное содержание программы по дизайн-мышлению не уместится в эту книгу. Но бóльшая часть курсов предлагает комбинацию тем, уже упомянутых в этой главе, например:

- навыки дизайн-мышления;
- понимание нужд потребителя;
- разработка услуги/потока работ;
- создание прототипов и тестирование;
- факторы, которые нужно учитывать в бизнесе;
- измерение и оценка.

Если у вас в компании уже есть сотрудники, занимающиеся дизайном продукта, они могут обучить команды разработчиков тому, как начать мыслить и действовать, как они. Если же таких нет, то можно посетить курсы по дизайну продукта в местном колледже или университете. Многие из этих учреждений могут предложить адаптировать лекции для вашей компании. Даже если вы просто работник небольшой организации, интересующийся этой темой, можете найти онлайн-курсы по дизайн-мышлению сами.

Одна компания, с которой мы разговаривали (крупный потребительский банк), решила создать свой курс по внутреннему дизайн-мышлению, где специалисты по дизайну продукта будут проводить занятия с командами API в офисах компании. Впоследствии команды API могли обращаться к своим наставникам за советом по улучшению дизайна API. Их целью не было превращение всех раз-

работчиков и архитекторов ПО в профессиональных дизайнеров. Они стремились к тому, чтобы команды API лучше понимали процесс дизайна и научились применять эти навыки в работе.

Важно помнить, что результаты дизайн-мышления — это больше, чем просто улучшение удобства использования или эстетической привлекательности ваших API. Это может привести к лучшему пониманию целевой аудитории (клиентов), созданию API, соответствующих конкурентным бизнес-целям, и более надежному процессу измерения успеха API в экосистеме.

Как бы ни был важен дизайн в подходе AaaP, это лишь начало. Важно также обращать внимание на первоначальный опыт работы с клиентами, после того как API будет выпущен и доступен для использования. Об этом мы поговорим в следующем разделе.

Поддержка новых клиентов

Любой, кто покупал что-либо у Apple, знает, что распаковка их продуктов может стать незабываемым опытом. И это не случайно. Многие годы у Apple существует специальная команда, занимающаяся только улучшением впечатления от распаковки.

Адам Лашински, автор *Inside Apple* («Внутри Apple»), пишет: «Несколько месяцев дизайнер упаковки сидел в своей комнате и занимался самым прозаическим делом — открывал коробки»[1]. Он продолжает:

> Apple всегда стремится использовать коробки, вызывающие идеальный эмоциональный отклик при открытии... Дизайнер создавал и тестировал бесконечную серию стрелок, цветов и лент для крошечной вкладки, предназначенной, чтобы показать потребителю, в какую сторону тянуть невидимую наклейку без рамок на верхней части коробки нового iPod. Он был просто одержим поиском идеального варианта.

И это внимание к деталям выходит далеко за рамки простого открытия коробки и извлечения устройства. Apple позаботилась о том, чтобы батарея была полностью заряжена, чтобы клиенты могли «запуститься и работать» в течение нескольких секунд, а общее впечатление было приятным и плавным. Команды

[1] *Condliffe J.* Apple's Packaging Is So Good Because It Employs a Dedicated Box Opener («Упаковка Apple настолько хороша, потому что в ней используется специальный открыватель для коробок») // Gizmodo, 25 января 2012 г. https://oreil.ly/JrY6S.

Apple по разработке продукта хотят, чтобы клиенты *полюбили* этот продукт с самого начала.

Стефан Томке и Барбара Фейнберг написали в «Случае из Гарвардской школы бизнеса»: «Стремление вызвать у людей любовь к нашим устройствам и их использованию всегда вдохновляло — и продолжает по сей день — разработку продуктов Apple»[1].

КОГДА API — ВАШ ЕДИНСТВЕННЫЙ ПРОДУКТ

Stripe — это успешная платежная система на основе отличного API, который высоко ценят разработчики. Оценка стартапа на 2021 год составила около 95 миллиардов долларов США при численности сотрудников менее 4000 человек[2]. Вся бизнес-стратегия его основателей заключалась в осуществлении платежных услуг через API. Поэтому они решили с самого начала инвестировать в дизайн-мышление и подход «API как продукт». Для Stripe API — *единственный* продукт. Отношение к API как к продукту помогло им достичь как технических, так и бизнес-целей.

Такое внимание к первому опыту использования продукта применимо и к API. Кажется, почти невозможно добиться того, чтобы разработчики их *полюбили*, но исполнение этой идеи влечет за собой далеко идущие последствия. Если ваши API сложны для понимания, разработчикам будет тяжело с ними работать, и если это займет «слишком много времени», они просто рассердятся и все бросят. В мире API время до того, как все заработает, часто называют «время до первого Hello, World» (Time to first Hello, World). В сфере онлайн-приложений его иногда называют «время до “вау!”» (Time to Wow!) — TTW.

Время до «вау!»

В статье «Ускорение роста: создание “вау-момента”» Дэвид Скок, сотрудник инвестиционной компании Matrix Partners, описывает важность «вау-момента», которого надо добиться в любых отношениях с клиентом: «“Вау!” — это момент... когда ваш покупатель вдруг видит выгоду, которую получает от использования продукта, говоря себе: “Вау! Это круто!”»[3]. И хотя Скок обра-

[1] *Thomke S. H., Feinberg B.* Design Thinking and Innovation at Apple («Дизайн-мышление и инновации в Apple»), пересмотрено в мае 2012 г. https://oreil.ly/wY4IA.

[2] *Lunden I.* Stripe Closes $600M Round at a $95B Valuation («Stripe закрывает раунд на сумму 600 миллионов долларов при оценке в 95 миллиардов долларов») // TechCrunch, 14 марта 2021 г. https://oreil.ly/60fbh.

[3] *Skok D.* Growth Hacking: Creating a Wow Moment («Ускорение роста: Создание вау-момента») // For Entreprenuers (блог), 2013. https://oreil.ly/YyVlI.