

25 июня 2024

5 этапов разработки ИТ-системы: от идеи до проекта

Создание ИТ-системы — многостадийный процесс: вначале формируется идея продукта, затем она трансформируется в концепцию решения, которая в итоге становится основой для создания системы. От того, насколько тщательно компания проработает эти стадии, напрямую зависит, будет ли результат соответствовать ожиданиям.

В 2022 году рынок заказной разработки в России вырос на 20% — бизнес активно «импортозамещал» ушедшие с рынка ИТ-продукты, а отечественные вендоры не всегда могли предложить готовые решения. В 2023 году, по оценкам экспертов ГК «КОРУС Консалтинг», тенденция сохранилась. Многие компании стали не только работать над импортозамещением критичных систем, но и инвестировать в цифровые инструменты для повышения конкурентоспособности.

Есть нюансы, которые важно учесть в начале работы над ИТ-системой, часто недооценивают или вовсе упускают из вида. Рассмотрим основные моменты.

ПРОВЕРКА ИДЕИ



Итак, у вас есть идея продукта. Прежде всего ее нужно протестировать. Очень важно понять, есть ли у идеи потенциал: позволит ли решение увеличить прибыль, сократить расходы или каким-либо другим способом повлиять на эффективность компании. Проверка идеи включает четыре основных шага. Они одинаково актуальны и для B2B, и для B2C-продуктов.

Рассмотрим конкретный пример. Разработка системы автоматизации программ лояльности для B2B-клиентов. Предположим, что производитель товаров повседневного спроса продает их через дистрибьюторскую сеть розничным магазинам и компаниям HoReCa. Также несколько федеральных сетей покупают продукцию бренда напрямую.

Шаг 1

Необходимо описать проблему, с которой столкнулась компания, и рассмотреть доступные способы ее решения вне области ИТ-автоматизации. Может выясниться, что проблему целесообразнее решить, например, увеличив штат, а создавать ИТ-систему экономически необоснованно. Этот шаг позволит вам проанализировать все пути решения бизнес-задачи и убедиться, что создание ПО необходимо, а инвестиции в него оправданы.

Проблемы, которые в нашем кейсе производитель хочет решить с помощью системы:

- отсутствие инструментов повышения лояльности торговых представителей и сотрудников отдела закупок компаний-клиентов;
- недостаток рычагов, которые стимулируют продажи определенных товарных групп в сегменте B2B.

Компания пробовала снижать цены на определенные группы товаров, однако такой подход не позволял сделать вывод об эффективности промоакции. У бизнеса не было системы, которая могла бы собирать данные о покупках клиентов и выявлять степень влияния промоакций на увеличение продаж.

Шаг 2

Важно пообщаться с будущими пользователями решения и выяснить, действительно ли проблема, которую оно должно решать, актуальна и значима для них. Часто мы наблюдаем ситуации, когда компания создает дорогостоящие сервисы, которыми пользуются несколько человек, не принося бизнесу ощутимых эффектов.

Оказывается, что в системе изначально была заинтересована лишь часть целевой аудитории, но на старте проекта руководство этого не знало. Чтобы инвестиции в разработку не были напрасны, важно установить, есть ли в решении реальная потребность. Кроме того, интервью с пользователями системы позволяют в разы сократить количество ошибок при реализации сервиса и сэкономить бюджет проекта.

В нашем примере интервью с сотрудниками, отвечающими за B2B-продажи в компании, показало, что систему поощрения важно сделать максимально прозрачной, — это обеспечит доверие клиентов. Кроме того, удалось выяснить, что для тех, кто работает в отделе продаж в компании-производителя, существуют материальные поощрения в виде бонусов, но они не обеспечивают сфокусированность сотрудников на тех или иных товарных группах.

Шаг 3

Необходимо проанализировать уже существующие ИТ-инструменты для решения проблемы. Если компания создает собственный продукт и планирует продавать его другим игрокам рынка, это поможет оценить конкурентов и понять, в чем их преимущества. А если система предназначена для внутреннего использования, этот шаг позволит рассмотреть все варианты решения бизнес-задачи. Может оказаться, что продукт уже есть на рынке и вам не обязательно создавать его с нуля.

Вернемся к нашему примеру — созданию системы для автоматизации программ лояльности. Анализ рынка показал, что для повышения лояльности заказчиков существуют облачные решения маркетинговых агентств, но такие системы обладают значительными недостатками: подобные решения не поддерживают необходимые компании механики для начисления бонусов, очень сложно дорабатываются. Кроме того, компания начинает зависеть от вендора — он берет на себя все задачи по развитию и поддержке системы.



Шаг 4

Сформулируйте гипотезу, как решение поможет компании заработать или сэкономить. Мы советуем формулировать гипотезу по принципу SMART — постановка цели в формате конечного результата — то есть так, чтобы ее можно было подтвердить или опровергнуть. Хорошая гипотеза всегда звучит четко и включает измеримые показатели.

В нашем примере гипотеза может быть такой: создание системы для автоматизации программ лояльности в сегменте B2B позволит увеличить объем продаж на 4% в следующем году.

Таким образом, первый этап показывает, насколько идея будущей системы перспективна. Если вы убедились, что задумка имеет потенциал, по окончании этой стадии у вас будут четко сформулированы ожидания от создания решения и вы сможете перейти к следующему этапу. А если идея проверку не прошла, эта фаза поможет уберечь от ненужных инвестиций.

РАЗРАБОТКА КОНЦЕПЦИИ РЕШЕНИЯ

После того как вы поняли, что идея имеет потенциал, и сформулировали, какого результата вы хотите достичь с помощью системы, можно переходить к деталям. На втором этапе нужно определить, как будет выглядеть решение: проработать UX, уточнить функциональность системы и механики, которые будут лежать в основе ее работы, а также спроектировать архитектуру

решения и понять, с какими системами необходимо будет его интегрировать.

В нашем примере на данном этапе мы определяем, что система автоматизации программ лояльности должна быть настраиваемая и гибкая, чтобы производитель мог применять различные механики и управлять целевой аудиторией, опираясь на динамику показателей продаж разных товарных групп.

Пользователи системы, менеджеры отдела закупок компаний-клиентов, в рамках проведения акции будут выполнять разные задачи: проходить обучение, покупать или продавать продукцию компании, оставлять о ней отзывы и за это награждаться призами. Помимо этого, наше решение должно получать данные из CRM, приложения клиента или от сотрудников, которые вводят такую информацию вручную, а затем передавать ее в системы класса ERP и Distribution management system.

ФОРМУЛИРОВАНИЕ ТРЕБОВАНИЙ

На следующем этапе формулируются функциональные и нефункциональные требования к будущей системе. Они описывают прообраз решения на бумаге.

Значимость этого этапа очень часто недооценивается. При этом некорректно составленные требования могут существенно повлиять как на ход проекта, так и на полученный результат: на основе требований формируется оценка стоимости и сроков проекта, а также принимаются решения, связанные



с архитектурой системы.

Кроме того, конкретизация параметров будущего решения поможет объективно оценить его реализацию и избежать разбега в оценках стоимости проекта от потенциальных подрядчиков в несколько раз.

Мы убедились в этом благодаря эксперименту: несколько команд оценивали реализацию одной и той же системы по очень короткому и общему описанию. Нас удивило, что разброс в оценках был колоссальным — разработчики абсолютно по-разному интерпретировали запрос. Это связано с тем, что при высокой степени неопределенности команда придумывает свой вариант реализации системы и в простом, на ваш взгляд, инструменте может предусмотреть обширную функциональность.

Помимо этого, при нечетко сформулированном запросе разработчики часто закладывают в оценку риски, связанные с тем, что создание системы потребует больших трудозатрат, чем можно предположить на первый взгляд, и рассчитывают стоимость проекта исходя из самого сложного сценария реализации.

К нам часто обращаются компании с просьбой провести предпроектную подготовку. Обычно наши аналитики вместе с клиентами формулируют требования на виртуальной доске. Это позволяет всем находиться в едином информационном поле.

Аналитик вместе со стейкхолдерами проекта составляет User Story Mapping, где описывает функциональное наполнение системы. User Story Mapping — это комплекс схем (User Story), каждая из которых описывает свой блок процессов. Чтобы представить их в максимально наглядном виде, мы разбиваем каждую User Story на три сущности: пользователь, интерфейс и действия.

В ходе интервью со стейкхолдерами мы выясняем, какие пользователи взаимодействуют с системой и что они должны уметь делать. Перед встречей со стейкхолдерами аналитик добавляет на User Story уже известную информацию, а во время беседы дополняет ее стикерами.

Рассмотрим User Story Mapping для системы автоматизации программ лояльности. Маркетологу необходимо иметь возможность настраивать программы лояльности. Ему должны быть доступны просмотр списков программ лояльности, их фильтрация по дате проведения, статусу, ответственному. Также маркетологу нужно иметь возможность открыть существующую или создать новую программу лояльности.

При составлении User Story Mapping важно не забывать о функциональности для авторизации и регистрации пользователей, ведения нормативно-справочной информации, фильтрации и сортировки списков и выполнения CRUD-операций: создание, просмотр, обновление, удаление. Нередко эти блоки упускают из вида.

Далее мы дополняем схему всеми необходимыми нефункциональными требованиями, например, к производительности и безопасности. Затем аналитик переводит визуализированные требования в двумерную таблицу, которую можно направлять на оценку команде проекта и в дальнейшем использовать на старте реализации проекта.

Ниже показали, как может выглядеть таблица в кейсе, который мы рассматриваем.

Важно также не забыть про требования к интеграциям и нефункциональные требования. Эти блоки требований значительно влияют на оценку, состав работ и на архитектуру системы.

Например, в нашем кейсе одно из требований к интеграциям может звучать так: «система должна ежедневно загружать и обновлять список внутренних пользователей из Active Directory». Пример нефункционального требования к системе автоматизации программ лояльности — «обеспечение защиты от перехвата данных при передаче их между клиентом и сервером через SSL-шифрование».

Важно, чтобы формулировкой требований занимался штатный или внештатный аналитик.

На это есть несколько причин:

1. Такой специалист задаст неочевидные вопросы. В нашем кейсе это может быть вопрос «Что делать, если сотрудник дистрибьютора уволится или

перейдет к другому дистрибьютору: начислять ли бонусы, дарить ли подарки?»

2. Он предложит самые оптимальные идеи реализации проекта с точки зрения соотношения затрат и удобства пользования системой. Например, в нашем кейсе аналитик может предложить создавать программы лояльности не в перегруженной элементами форме, а в пошаговом помощнике или посредством заполнения атрибутов в карточке.

3. Наконец, аналитик учтет требования законодательства к хранению персональных данных, ИБ, особенностям налогового учета.

Следующий шаг — определить, кто будет заниматься проектом.

ВЫБОР СПОСОБА РЕАЛИЗАЦИИ ПРОЕКТА

Компания может взять разработку на себя, подключив к ней своих сотрудников. Другой вариант — найти подрядчика. Он предоставит команду разработки и возьмет на себя управление проектом и ответственность за весь процесс.

Помимо этого, можно прибегнуть к аутстаффингу — в этом случае основная работа выполняется сотрудниками компании-заказчика, но для отдельных задач привлекаются специалисты подрядчиков. Управление проектом осуществляет заказчик. Другой комбинированный подход — когда к внешней команде, которой руководит подрядчик, присоединяется несколько сотрудников компании-заказчика. В этом случае за управление проектом и его результаты отвечает подрядчик.

Каждый вариант имеет право на существование и обладает своими плюсами и минусами. При выборе варианта реализации важно учесть особенности вашей компании, доступность внутренних ресурсов, бюджет и сроки проекта.

ПОИСК ПОДРЯДЧИКА

Очевидно, что когда вы решаете разрабатывать систему собственными силами, выбирать подрядчика не надо. В случае аутстаффинга нужно запросить у потенциальных кандидатов ставки и CV специалистов, а потом провести с ними собеседования. Важно обращать внимание не только на hard, но и на soft skills.

Другой вариант — когда проект выполняется подрядчиком под ключ.

Здесь алгоритм действий может быть таким:

- 1.** Определите пул потенциальных партнеров для реализации проекта.
- 2.** Проведите исследование рынка возможных подрядчиков: посмотрите их портфолио, соберите отзывы о компаниях от коллег и в ИТ-сообществе, проверьте репутацию подрядчиков, их присутствие в инфополе, награды и рейтинги, проанализируйте надежность подрядчиков, включая их финансовое состояние и способность выполнить проект в срок.
- 3.** Направьте потенциальным партнерам Request for proposal (RFP), включив в него требования к системе, которые прорабатывали на предыдущем этапе.

- 4.** Оцените, обладает ли партнер отраслевой экспертизой. Если подрядчик задает вопросы, которые показывают, что команда не понимает, как устроена ваша индустрия, это тревожный сигнал.
- 5.** Обратите внимание на то, стремится ли партнер погрузиться в задачу, вникнуть в особенности процессов компании и разобраться в нюансах того, как система будет использоваться. Задайте себе вопрос, комфортно ли вам общаться с командой подрядчика.
- 6.** Получите коммерческие предложения и сравните их. Обратите внимание на то, насколько проработана оценка проекта: детализация — признак зрелости подрядчика.
- 7.** Обсудите полученное предложение с подрядчиками. После того, как вы поймете, что включила в оценку каждая из команд, вы сможете объективно оценить проект и выбрать оптимального поставщика услуги. Для этого мы используем Excel-таблицу, которая позволяет на основе оценок, полученных от подрядчиков, понять длительность проекта и необходимое для работы количество сотрудников.
- 8.** Познакомьтесь с потенциальным менеджером проекта и командой. Это позволит проверить их квалификацию и слаженность работы друг с другом.

Не обязательно через все эти стадии проводить каждого кандидата — тех, кто не справился с тем или иным этапом, можно сразу отсеять.

Чек-лист выбора подрядчика

1. Понимание индустрии заказчика и ее особенностей.
2. Квалификация выбранного проектного менеджера и его команды, ее сработанность.
3. Опыт подрядчика: кейсы, клиенты, награды.
4. Финансовая устойчивость.
5. Простота коммуникации — вы говорите на одном языке.
6. Заинтересованность в вас как в клиенте и желание погружаться в задачу и проект.

Дополнительный лайфхак, который поможет выбрать подрядчика: попробуйте сначала поработать вместе с потенциальным партнером над небольшой задачей, а потом примите решение о том, стоит ли вместе реализовывать крупный проект. В качестве такого «тест-драйва» мы часто предлагаем нашим клиентам подготовить требования к будущей системе. Это дает компании-заказчику возможность не только проработать параметры решения и подготовить документацию для тендера, но и составить впечатление о команде, уровне ее профессионализма и оценить, насколько комфортно будет с ним работать.

ЗАКЛЮЧЕНИЕ

Аналитика и качество проработки требований к будущей ИТ-системе на старте проекта напрямую влияют на конечный результат и экономический эффект от внедрения решения. Тщательная подготовка к созданию системы поможет бизнесу застраховать себя от множества рисков, начиная от выхода за рамки бюджета проекта и заканчивая необходимостью менять подрядчика в процессе разработки.

Именно поэтому важно заранее предусмотреть нюансы, о которых мы рассказали выше — они позволят избежать большинства ошибок при реализации системы и выбрать подходящую для этого команду.