

Как подготовить бэклог продукта с большим количеством зависимостей и не потратить время впустую / Хабр

Привет, меня зовут Макс, я продукт команды Self-Service в мобильном приложении Тинькофф. У моей команды три основные цели по созданию сервиса: contactless, proactive и self-service.

Это значит, что мы стараемся сделать незаметными процессы для пользователя: убрать то, что можно убрать, и сделать незаметным то, что можно сделать незаметным. Проблемы решаем проактивно, если можем предсказать проблему, и упрощаем процесс за счет технологий.

Сегодня хочу рассказать о том, как мы в команде подходим к оценке задач и их приоритизации. А еще о том, как не делать лишнюю работу в условиях сложного продукта с большим количеством зависимостей.

Если вы отвечаете за разработку продукта, то в процессе изучения конкурентов, качественной и количественной аналитики, общения с коллегами или стейкхолдерами у вас появляются идеи, да и желающих накинуть их всегда хватает.

Представим, что вы всё записываете в заметки или блокнотик. У здорового продукта возникает логичный вопрос: какие задачи решать в первую очередь?

Главная задача хорошего бэклога — снизить уровень неопределенности и дать пользователю как можно больше ценности за меньшее время. Главное здесь — не ставить на первый план только цифры. Иначе за год у вас не появится ничего, кроме бэклога, если продукт большой и сложный. А через год бэклог уже устареет.

Раньше я использовал классическую ICE-методику, но в Тинькофф пришлось ее адаптировать. Сейчас расскажу как.

Мои пункты оценки бэклога

- Название идеи — краткое название и суть того, что хочется сделать.
- Основная гипотеза — как фича повлияет на пользователей и бизнес. Например: если мы добавим фичу Изменить номер в МБ, то сократим обращения к оператору, время до целевого действия и количество касаний. То есть, количество шагов нужное пользователю, чтобы решить вопрос, сократится: мобильное приложение — бот — первая линия — бэк-офис и т. д. Форматы описания гипотезы есть в свободном доступе, их перечислять я не буду. Главное, чтобы вы и команда понимали, что делаете и зачем.
- Ожидаемый результат — что должно получиться в итоге. Описание должно содержать критерии приемки задачи, а результат — конкретику. Например, после реализации фичи пользователь может изменить контактный номер телефона без обращения в банк.
- Вариант MLP/MVP. MVP — уже знакомое понятие, но недавно появилось еще одно — MLP: minimal loveable product. Уверен, вы и раньше использовали MLP, просто не называли его так. Важно подумать про то, как мы проверим нашу гипотезу и получим первые инсайды до того, как потратим ресурсы на целевое решение. Стоит использовать MLP, если финальное решение не зависит от результатов A/B-теста, есть возможность отдельно реализовать сценарий в MLP и вы уже сейчас можете поставить основной сценарий на прод без потери качества. Результаты MLP помогут

вам в приоритизации задач, добавят инсайтов и аргументов для правильного распределения ресурсов. Подробнее почитать про MLP можно в первоисточнике или на русском языке. Критерии хорошего бэклога Для определения приоритетов я использую ICE-метод. В ICE мы оцениваем идеи по трем параметрам: impact, effort и confidence. Impact показывает, насколько идея положительно повлияет на ключевой показатель, который хотим улучшить. Effort — оценка того, сколько усилий и ресурсов требуется для реализации этой идеи. Confidence демонстрирует, насколько мы уверены в оценках влияния и легкости реализации.

Подробнее об impact

Чтобы определить влияние на метрики — составляем справочник. Мне нравится подход, в котором за единицу impact берется 1% от значения основных продуктовых метрик. Например, ваша основная продуктовая метрика — это САС, и она равна 1500. Impact в этом случае будет равен 15 (1% от 1500).

Хорош, если продуктовые метрики будут соответствовать трем основным критериям:

- Показывать ценность продукта для потребителя.
- Пересекаться с целями компании.
- Понятно влиять на деньги.

В моем случае главная продуктовая метрика — доля активных клиентов мобильного приложения, которые не обращаются в поддержку.

Метрика верхнеуровневая и нечувствительная. Она не подходит для составления справочника, оценки работы результатов команды и отслеживания результатов эксперимента. Поэтому декомпозицию ее на более приземленную: доля активных клиентов мобильного приложения, которые не обращаются в поддержку по определенной тематике. За значение impact 1 я беру 10 000. Топ обращений не волатилен в течение года в относительных величинах. Под значениями 1—10 можно подразумевать что угодно — главное, чтобы значения были согласованы между собой.

Оценивая гипотезу, я выбираю соответствующее значение. Например, фича изменение номера в мобильном приложении сократит обращения к оператору по тематике изменение данных на N в месяц. Подставляем соответствующее значение impact.

В данном кейсе определить это число просто. Нам известно поведение пользователей. Сколько человек пишут в чат, чтобы изменить номер; сколько заходят на экран профиля, перед тем как обратиться; у какой доли обратившихся есть мобильное приложение и т. д. Но так бывает не всегда. Если вы хотите побольше погрузиться в то, как считать потенциальный impact, рекомендую книгу Как измерить все, что угодно Дугласа Хаббарда.

Позже каждая фича проходит через A/B-тест, где основные метрики для оценки эффективности — количество/доля обращений по тематике, время до целевого действия и количество касаний.

Подробнее о том, как выбрать основные продуктовые метрики, расскажу в следующей статье.

Подробнее о confidence

Составляем справочник по типу выше. Можно провести опрос и понять, как коллеги доверяют тому или иному факту. Еще тут хорошо работает логика. Если нужен пример того, как можно логически определить уверенность для гипотезы, спрашивайте в

комментариях.

Чтобы ваша логика была понятна другим и не порождала много споров, можно составить справочник.

Подробнее про effort

Собираемся с теми, кто будет участвовать в разработке фичи, и пишем ожидаемое время разработки в часах/днях/неделях. Главное — использовать одну единицу измерения. Чем больше у вас будет информации по задаче к этому времени, тем точнее будет оценка.

С ICE все — дальше только ссылка на трекер с задачей: с подробным описанием идеи, дизайном, ссылкой на исследование и т. д.

В чем подвох описанного выше метода в реалиях нашей команды? Хороший продукт — это всегда сто один компромисс и умение управлять ими.

Продукт нашей команды — сервис. Сервис проходит через все бизнес-линии, продукты и процессы. На оценку и формирование бэклога может уйти немало времени.

Допустим, для оценки effort желательно иметь подробное описание задачи и дизайн, а это как минимум ресурс продакта, аналитика, дизайнера, коллег с бэка и других бизнес-линий. Их время можно потратить впустую, потому что может оказаться, что задача нереализуема в ближайшие полгода или год. Дизайн и аналитика устареют — работа проделана зря.

Из-за этого мои гипотезы проходят некий прескоринг перед тем, как попасть в бэклог. Его задача — быстро найти ограничения, постараться понять, где нужно решать проблему, и отправить на оценку в бэклог реализуемые задачи, а не мысли звездочета.

Не каждой команде нужны все пункты ниже, но какие-то взять на вооружение точно можно. В книге *Вдохновленные*: все, что нужно знать продакт-менеджеру Мартина Когана описана часть ограничений. Ваша задача — как можно раньше понять, что может помешать вам выполнить вашу задачу, выписать эти причины-критерии и проскорить фичу по ним.

Мой прескоринг

Расскажу, что включает в себя мой прескоринг.

Бизнес-ограничения. Почкинив что-то в одном месте, можно сломать в другом и сделать в итоге хуже, чем было. Не все фичи нужно делать. Приходите к нашей команде, чтобы узнать, как фича может повлиять на клиентский опыт: негатив, обращаемость, отзывы и т. д.

Технические ограничения. Нужно убедиться, что фича реализуема в ближайшие 3—6 месяцев. При выборе периода исходим из скорости движения вашей команды. Это нужно для того, чтобы правильно определить приоритеты и делать то, что не заблокируется другими системами.

Дизайн-ограничения. Важно, чтобы фича не дублировалась с другой командой и на экране, где планируются изменения, не будет редизайна.

Поиск заинтересованных сторон aka бизнес-возможности. У наших задач много зависимостей от бэка, они затрагивают больше 10 команд. У этих систем много заказчиков и свои приоритеты. Чтобы задача попала в их бэклог с правильным приоритетом, нужно

комплексно оценить ее. Возможно, вашей фичей вы положительно повлияете не только на метрики вашей команды, но и — о чудо! — на метрики других команд. Это очень важно, потому что поможет правильно оценить value и заручиться поддержкой других продуктов.

Эксперт по процессу. Я записываю всех, кто может проконсультировать по тому или иному вопросу. Чтобы был понятен масштаб: в моей табличке уже больше ста человек. Это можно проделывать в голове или красиво оформлять в таблицу — все зависит от размера продукта и количества зависимостей.

Пример прескоринга фичи изменение контактного номера в мобильном приложении

Бизнес-ограничения. Явных ограничений нет, но нужно понаблюдать за метриками других продуктов, чтобы выяснить, как мы на них влияем.

Технические ограничения. Сама возможность изменить номер в мобильном приложении доступна не всем типам клиентов, а значит, нет смысла показывать ее тем, кто воспользоваться ею не сможет. Не подходит бэк для реализации фичи в мобильном приложении, так как текущий бэк создавался для операторов колл-центра.

Требуемые доработки в бэке:

- Метод смены контактного номера телефона в сервисе изменения клиентских данных Person Profile.
- Доработка сброса кэшей в API.
- Рефакторинг многошаговой конфирмации в API.
- Доработка сервиса лицевой биометрии.
- Доработка конфирмации в core-компоненте.

За каждый пункт отвечают разные команды. Нужно пообщаться с каждой, чтобы понять сроки и запланировать задачу.

Требуемые доработки в чат-боте. Для разных типов клиентов нужен свой, особенный ответ чат-бота: одних ведем на экран в МП, вторых в КЦ.

Дизайн-ограничения. Планируется редизайн экрана другой командой, где будет точка входа в фичу — синк с командой, ответственной за экран.

Поиск заинтересованных сторон. Чтобы лучше спланировать бэклог и более точно посчитать велью фичи, я выделил четыре команды.

Команда origination: изменение контактного номера в МП может заэфектить долю клиентов с актуальным номером телефона, что влияет на скорость обработки заявки и конверсию из заявки в открытие продукта.

Команда person profile: чем выше доля актуальных контактов, тем ниже расходы на их дедупликацию.

Команда Тинькофф Мобайла: один из продуктов экосистемы Тинькофф — мобильный оператор Тинькофф Мобайл. Как правило, контактный номер телефона в банке основной для клиента. NPV основного номера выше на X. Фича изменить контактный номер в МП поможет промотировать, сделать номер Тинькофф Мобайла контактным в банке, увеличить NPV Мобайла и не увеличить расходы на обработку обращений.

Экосистема: с фичей появится возможность делать новые рекомендации. Сейчас в тесте два кейса: клиенту не придется вводить новый контактный номер, так как мы уже его знаем, и клиенту не придется даже искать фичу изменить номер, так как мы знаем, что он пришел его изменить (онлайн-триггеры), и можем предложить это раньше, чем он начнет искать, как это сделать, или задаст вопрос в чате.

Заключение

- Прескорьте свои гипотезы до того, как они попадут в бэклог. Так можно сэкономить время, особенно в большой компании с множеством неизвестных.
- Если ваш бэклог не отвечает на вопросы Что, зачем и в каком порядке мы делаем? — это не бэклог. Вам и членам вашей команды должно быть сразу понятно, почему вы делаете фичу А после фичи Б.
- Если задачи выполняются, а метрики не меняются — значит, вы движетесь не туда.
- Гайды и статьи в интернете — не истина в последней инстанции. Исходите из особенностей вашего продукта и той среды, где находитесь.

Шаблон бэклога - backlog

- product management
- backlog
- приоритизация задач
- Блог компании Т-Банк
- Разработка мобильных приложений
- Аналитика мобильных приложений
- Управление продуктом