



Как преодолеть пропасть  
между бизнесом и  
разработкой

Немцан Михаил  
Product manager  
mnemtsan@iiko.ru



# О компании iiko

- Компания занимается комплексной разработкой ПО для автоматизации отрасли HoReCa (Hotels, Restaurants, Catering).
- Работу над продуктом начали в 2007-м году.
- На текущий момент в компании работает более 100 человек.
- Продуктом пользуется более чем 7000 ресторанов в СНГ, Юго-Восточной Азии, Западной и Восточной Европы.
- Основной объем продаж делается компаниями-партнерами.
- Продукт коробочный.



# Проблемы, с которыми мы сталкивались

- Передача требований через несколько звеньев (клиент – партнер – менеджер по работе с партнером – аналитик – разработчик), потеря исходных требований.
- Плохое юзабилити, недоделанный функционал с точки зрения клиента (с точки зрения разработчика все в порядке).
- Плохая фокусировка, фичи «в стол».
- Долгая верификация доработок.
- Принятие требований одного клиента за «требования рынка», в результате другие клиенты считают что доработка сделана «для марсиан».
- Другие страны – другой бизнес. То что работает в России, не всегда находит понимание в Европе и странах ЮВА.



# А давайте посчитаем (наш сценарий)

- Зарплата разработчика в день (с налогами и затратами на аренду) ~ 5300, аналитика ~ 4800, тестировщика ~ 4 000, тех.писа ~ 3700
- Стоимость разработки средней задачи составит ~ 15 (разработка) \* 5300 + 2 \* 4800 (аналитика) + 5 (тестирование) \* 4000 + 2 (описание) \* 3700 = 116 500 – примерная цена ошибки понимания требований клиента. А если посчитать сколько будет стоить поддержка неправильного решения, цена станет совсем запредельной.
- Средняя цена автоматизации ~ 80 000, с учетом того, что большая часть уходит партнеру-автоматизатору, мы должны продать наш продукт 4-6 раз, чтобы отбить стоимость ошибки. При таком рынке и такой конкуренции подход «а давайте сделаем и посмотрим как отреагирует клиент» может привести к банкротству ).



# Как мы решали эти проблемы

- Найм правильных людей.
- Убираем лишние звенья в цепочках передачи требований.
- Работа с экспертами в предметной области.
- Образование в сфере бизнеса.
- Особый подход к выпуску версий.
- Юзабилити-тестирование.
- Работа с мотивацией сотрудников.



# Найм сотрудников

- Лучше не взять хорошего человека чем ошибиться и взять плохого.
- Нанимать самому, не доверять HR-м.
- Брать только тех людей, чей темперамент совпадает с темпераментом команды/продукта.
- На собеседовании нужно спрашивать про проекты, смотреть как сотрудник их описывает.
- Брать только тех, кто вникал в проект целиком – старался понять для решения каких задач делался проект. Избегать найма «технических исполнителей».
- Давать на собеседовании задачи в терминах предметной области, смотреть какие вопросы человек задает по уточнению постановки задач.
- «Проверка боем» - в первый же месяц дать сотруднику задачу по реализации продуктовой доработки по всей системе в целом.



# Выявление требований

- Убирайте лишние звенья в цепи передачи информации. Старайтесь выйти на контакт с клиентом напрямую.
- Выясните роль постановщика задачи. Люди как правило искажают исходные требования в зависимости от профессии и должности.
- Клиент приходит с запросом доработки – обязательно узнайте какую проблему он хочет решить.
- Всегда старайтесь найти решение исходя из текущих возможностей системы.
- Часто правильнее убедить клиента поменять процессы чем автоматизировать то, что он попросил (автоматизация хаоса – автоматизированных хаос).
- Прежде чем делать то, что попросил клиент постарайтесь проконсультироваться с другими пользователями системы и понять общая ли это проблема или частная. Как другие клиенты решают проблему?



# Разработка

- Рассказывайте разработчикам и тестировщикам исходную проблему, которую нужно решить. Нужно стараться чтобы сотрудники команды почувствовали эмпатию по отношению к клиенту.
- Ставьте задачи по тестированию с описанием пользовательских сценариев.
- После окончания кодирования необходимо провести юзабилити-тестирование доработки. Подходы к тестированию можно использовать разные (в зависимости от зрелости UX-процессов в компании), но нельзя упускать оценку удобства решения основных сценариев клиента.
- По возможности следует привлекать клиента к процессу предварительного просмотра задачи.
- По возможности следует делать новый функционал отключаемым и отключать его до тех пор, пока мы не получим подтверждение его работоспособности от широкого круга клиентов.
- Нужно поощрять стремление к качественной реализации функционала. Часто лучше отодвинуть срок сдачи, но сделать хорошо.





# Выпуск версии

- Нельзя набирать доработок по принципу «пошить 7 шапок из одной шкуры». Лучше отказать некоторым клиентам, чем сделать сырой функционал.
- Подумайте на какой проблеме продукта Вы хотите сделать фокус.
- Доработки должны быть сбалансированными и не содержать разомкнутых контуров.
- Обязательно заложите время на переделку UI по результатам юзабилити-тестирования на реальных сценариях.



# Обратная связь от клиентов

- Разработка не может эффективно сфокусироваться без понимания проблем клиентов. Даже самое подробное описание требований не заменит беседы с непосредственными пользователями продукта.
- Необходимо устраивать регулярные встречи всей команды разработки и клиентов.
- Нужно поощрять чтение профильной литературы по предметной области и посещение разработчиками и тестировщиками профильных конференций и выставок.
- Желательно периодически отправлять сотрудников «на фронт» - поработать в тех организациях, которые используют продукт.



# Мотивация сотрудников

- У команды должно быть осознание того, что они делают отличный, качественный продукт, который решает реальные проблемы клиентов и обеспечивает конкурентное преимущество их бизнесу. Это должно стать частью корпоративной культуры так, чтобы делать посредственно стало зазорно.
- Снисходительное отношение к качеству, продукту или к своей работе должно строго пресекаться вплоть до увольнения.
- Необходимо учитывать пожелания команды к продукту. Полезные идеи нужно ставить в разработку. Если этого не делать, то коллектив потеряет сопричастность с продуктом.
- Немного элитарности не повредит.
- Ошибки позволительны, если человек готов оперативно исправлять их последствия.
- Слова не должны расходиться с делом. Нельзя говорить про качественный продукт, но не давать людям его сделать путем зажимания сроков.



# Образование по предметной области

- Обсуждайте на стендапах сценарии использования продукта, интересные бизнес-задачи, конкретные примеры как при помощи продукта клиенты получили конкурентное преимущество.
- Отправляйте сотрудников на конференции, профильные по продукту.
- На встречах с клиентами ведите беседу так, чтобы сотрудники могли узнать новое про суть автоматизированного бизнеса.
- Проводите обучение внутри компании. Сотрудники отдела консалтинга могут проводить обучение сотрудников отдела разработки.
- Покупайте книги по предметной области, разбирайте на стендапах эти книги.



# Результаты

- Команда обладает следующими качествами – компетентная, боеспособная (в смысле нет зависимостей от других команд), знает предметную область, радеет за бизнес клиентов.
- Зависимость от ПМ-а сильно уменьшилась, по сути ПМ стал почти не нужен.
- Разработчики с полуслова понимают клиента.
- Практически нет текучки в команде, люди или быстро уходят (неправильные люди прошли через сито найма) или остаются надолго.
- Существенно снизилось количество случаев, когда мы неправильно решали задачу клиента.
- Мы научились предвосхищать пожелания клиентов и убеждать их менять процессы для лучшей автоматизации.



Вопросы?

