

Производство счастья промышленными методами

Анна Тарасенко, 7bits, Омск. 2012 год



<http://www.devconf.ru>

Для кого-то я буду выглядеть вот так:



Спецкурс в ОмГУ, используемые языки и фреймворки

- Java/Spring
- Groovy/Grails
- Python/Django
- Ruby/Rails
- Routing
- Domain model
- Multi-thread model
- Deployment

Ди Марко, Листер «Человеческий фактор»



Повышение эффективности труда



У меня есть кость, миска и большой двор, я знаю, что должен быть счастлив



Менеджер-мультиинструменталист



Страшный сон руководителя



**Почти ко всем докладчикам возникает один
вопрос:**

Ок, это все здорово, а какую задачу заказчика вы этим решили?

Проблема:

- Команда не желает вникать в проблемы заказчиков и пользователей
- Менеджмент и заказчики намеренно не информируют команду о многих аспектах решаемой задачи

Симптомы проблемы-1

- Заказчик: молчит о проблемах и рисках бизнеса
- Заказчик: при переговорах а старте CIMM или ISO важнее людей
- Разработчики: телепаты в отпуске
- Разработчики: после нас хоть потоп
- Менеджеры: разработчики как дети и их надо развлекать
- Разработчики: работа - рутина, обязательно нужно отвлекаться для сохранения мотивации

Симптомы проблемы-2

- Тонны документов, но никто не видел пользователя
- Человек человеку волк
- В чужом глазу связка соломы
- Они никогда ничего не дают вовремя
- Основное орудие пролетариата - сковородка для прикрытия тылов
- Любой agile-процесс превращается в формальный ритуал

Последствия для команды (в долгосрочной перспективе)

- Мотивация снижена до критической отметки
- Работаем лишь бы не влетело
- Без бумажки ты ...
- Тебе что - больше всех надо? Пользователь никуда не денется
- Текучка лучших кадров
- Единственной мотивацией становится денежная, карьерная и вовремя уходильная

Последствия для внутреннего менеджмента

- Нужны специальные меры по контролю за выполнением задач
- За долгий срок команду покидают наиболее инициативные, те, кто остается, уже не вывозят
- Мотивировать команду не допустить срыва можно только кнутом и пряником

Последствия для конечных пользователей

- Нет прямого контакта с исполнителями и быстрой обратной связи
- Потребности удовлетворяются не полностью, долгое специальное обучение
- Порядок и сроки выполнения заданий по добавлению нового функционала не соответствует потребностям пользователей

Последствия для менеджмента заказчика

- Заказчик многократно переплачивает за завышенные сроки и текучку кадров
- Качество продукта отличное только по бумагам, на деле - заказчик переплачивает за неудовлетворительное качество
- Уход пользователей

История № 1

Автоматизация работы отдела экологии в крупном институте,
проектирующем магистральные трубопроводы

Портрет отдела разработки



Допущенные ошибки

- Сбор требований посредством опросов и изучения экселевских файлов
- Выбор малоизученной разработчиком на тот момент технологии разработки, что затянуло появление первой версии
- Попытка сделать первую версию системы сразу мега-универсальной

Как следовало сделать

- Погрузиться в бизнес-домен вплоть до выполнения основных юскейсов самостоятельно на реальном проекте
- За 1-2 недели написать несколько макросов на Visual Basic, ускорив работу отдела примерно на треть
- Разработать настоящую систему по уже изученным юскейсам, ускорив разработку не менее, чем на треть

История № 2

Аналитическая система для финансистов и маркетологов для разработчиков мобильных приложений

Выводы из историй №1 и № 2

- Пользователя надо знать в лицо, причем не только менеджеру и бизнес-аналитикам, а всей команде
- Надо получить с пользователей информацию о 80% вариантов использования, их реализация займет 20% времени
- В процессе создания функциональных и интерфейсных требований по максимуму должна участвовать вся команда

История № 3

Система управления контентом для издательского бизнеса

Выводы из истории № 3

- Ограничить влияние технического менеджмента со стороны заказчика в пользу бизнес-менеджмента
- Задавать вопросы заказчику обо всех сроках и грядущих событиях
- Запускать продукт в альфа-тестирование как можно раньше, в идеале - проверять прототипы на конечных пользователях

История № 4

Система управления камерами видео-наблюдения в небольшой аутсорсинговой компании

Выводы из истории № 4

- Команда должна быть в курсе всех рисков и проблем бизнеса заказчика
- Команда должна быть в курсе мотивов собственной компании
- При обсуждении и выборе технических решений следует руководствоваться проблемами бизнеса
- Все игры с новыми технологиями - на своих внутренних проектах или в open source

История № 5

Долгосрочный проект в рамках крупной аутсорсинговой компании

Выводы из истории № 5

- В иерархической команде критическая масса людей, избегающих ответственности и риска
- На новых небольших проектах разработчики пришлые или беглые с других проектов
- На фоне остальных проектов они быстро теряют мотивацию, часто уходят в стартапы или фриланс
- Меньше уровней иерархии – выше доля счастья на душу населения
- Гибкость для заказчика в маленьких командах выше

Формула любви счастья

- Решаем важную проблему – прибыль приходит сама
- High level skill команда, энтузиастов и евангелистов
- Смотрим на проблемы конечных пользователей или заказчика с точки зрения business value
- Заказчик доверяет команде решать, каким образом обеспечить потребности бизнеса в рамках обозначенного бюджета
- Рутинная автоматизирована
- Проверяем гипотезы, прототипируем, проводим разбор полетов, обмениваемся опытом, бэкапим решения

Спасибо, кэп!

- Это круто, но как этого добиться?
- И что - все должны оставаться маленькими, но гордыми?
- А как же прибыль?
- А куда девать всю толпу говнокодеров, которая у нас уже есть?
- А что делать, если мы вечные субподрядчики?

Спасибо за внимание!

- annie.tarasenko@7bits.it
- annie.tarasenko@gmail.com
- @AnnieOmsk