# Вступление

Привет!

# Риски – это большая проблема

Риски:

1. Проваленный срок
2. Отмененный проект
3. Замена системы из-за непригодности в долгосрочном использовании
4. Огромное количество дефектов в готовом продукте
5. Созданный продукт не решает задач бизнеса
6. Бизнес изменился – пока делали полгода продукт, рынок поменялся
7. Большое количество задач было с точки зрения программирования, но не бизнеса
8. Люди уходят, не выдержав проекта

# Кривая затрат

Как меня учили в университете: делаем все последовательно.

Ошибка, исправленная на стадии формирования требований, будет стоить десятки рублей, та же самая ошибка, найденная на стадии тестирования, обойдется уже в десятки тысяч.

Мы должны согласиться с тем, что система будет изменяться со временем. Это неизбежно. Меняются требования пользователей, меняется бизнес.

Мы стремимся сделать кривую затрат пологой.

# Экстремальное программирование

Экстремальное программирование принимает то, что требования к системе меняются, за данность.

Почему «экстремальное»? Потому что все по максимуму.

# Игра в планирование

# Частые выпуски

# Тестирование

Как обычно происходит создание системы?

Мы проектируем систему, создаем ее, тщательно тестируем и отдаем заказчику.

А вот так происходит разработка в XP. Мы начинаем с тестов. Это обязательное правильно.

# Рефакторинг

Рефакторинг есть следствие эволюции системы, ее естественного развития

Сравнить с ремонтом. Можно забыть о подтекающем кране, о сломавшейся форточке, о плитке, которая отходит. Ремонт каждого из этих поломок длиться пол дня. Но в один прекрасный момент кухня дойдет до такого состояния, что в ней невозможно будет находится. И тогда вы отколупаете всю плитку, которая и так на ладан дышит, снимите окна, из которых стекла вылетают. Ремонт будет длиться неделю и при этом на кухне будет невозможно находиться.

# Парное программирование

В прямом смысле этого слова. Два программиста за одним компьютером.

Как бороться с сопротивлением:

* Не обязательно явно говорить о том, что с этого дня вы работаете только в парах

Почему я не хочу программировать в паре:

* Я хочу стать героем
* Боюсь продемонстрировать свои недостатки
* Я не уверен в том, что моя идея работает
* У меня плохое настроение
* Я люблю чувство медитации

# Экспресс-совещания

Никто не может знать абсолютно всего. Парное программирование частично решает эту проблему, но чем занимаются другие пары? У кого именно надо получить интересующую вас информацию?

Экспресс-совещания – коротное ежедневное собрание всех членов команды с целью обмена информацией о текущем состоянии дел.

Цель – соединить тех, кто владеет информацией и тех, кто в ней нуждается.

Каждый должен ответить на вопросы:

* Что я сделал вчера и что узнал нового?
* Как далеко я продвинулся в решении своей проблемы и какие возникли препятствия?
* Что я буду делать сегодня?

Причины, из-за которых собрания могут не проводиться:

* не набрать необходимого количества участников
* люди, обладающие информацией, не хотят ею делиться
  + информация наделяет их высоким статусом
  + боятся показать, что знают не много
* люди, обладающие информацией, пытаются сообщить присутствующим слишком много (углубляются в детали)
* нет места для проведения

# Постоянная интеграция

# Простой дизайн

Простой не значит плохой или некачественный.

# Заказчик в команде

# Приемочные тесты

# Метафора системы

# Коллективное владение кодом

Автор кода – вся команда

# Стандарты кодирования

# 40 часовая рабочая неделя

# Заключение

Если у вас будут эти навыки, вы будете ценными сотрудниками.