

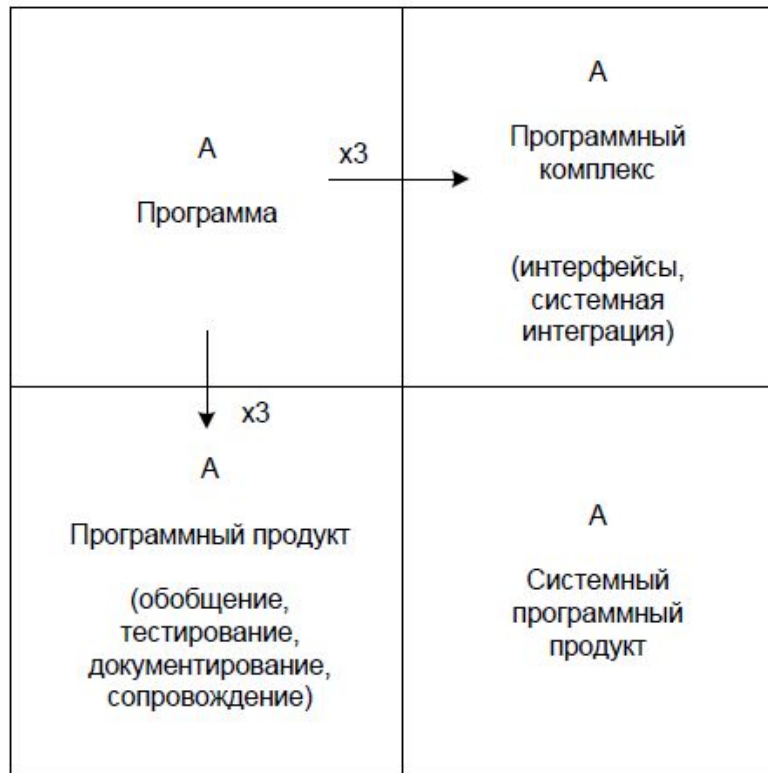
Software Engineering

Лекция 01: О разработке и разработчиках ПО

Тимофей Брыксин

timofey.bryksin@gmail.com

Программа и программный продукт



Программирование VS Промышленные проекты

- учебные задачи

- 4fun

- выпуск продукта

Контроль качества

работа в команде

Планирование

- за деньги заказчика

- сроки, результат, качество

Анализ и проектирование

Разработка требований

Сопровождение

Управление требованиями

Организация процесса

Документирование

Управление проектом

Программная инженерия

- особая область знаний/научная дисциплина
 - накопление опыта
 - фиксация лучших методов и практик
- конференция NATO Software Engineering, 1968
- кризис программного обеспечения
 - стоимость проектов превышает бюджет
 - превышаются сроки выполнения проектов
 - ПО слишком неэффективно
 - ПО имеет слишком низкое качество
 - ПО не отвечает необходимым требованиям
 - неуправляемые проекты, трудности с поддержкой кода
 - ...

Standish Group 2015 Chaos Report

MODERN RESOLUTION FOR ALL PROJECTS

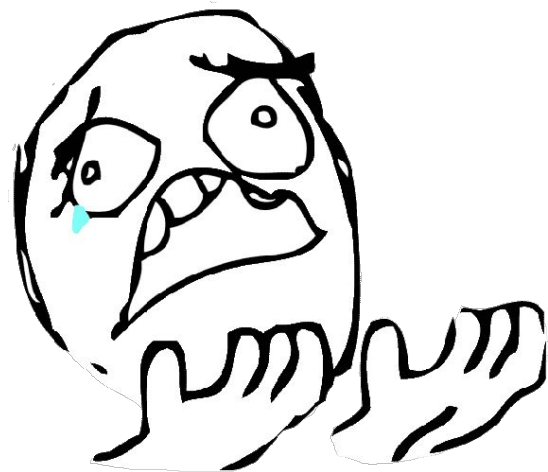
	2011	2012	2013	2014	2015
SUCCESSFUL	29%	27%	31%	28%	29%
CHALLENGED	49%	56%	50%	55%	52%
FAILED	22%	17%	19%	17%	19%

CHAOS RESOLUTION BY PROJECT SIZE

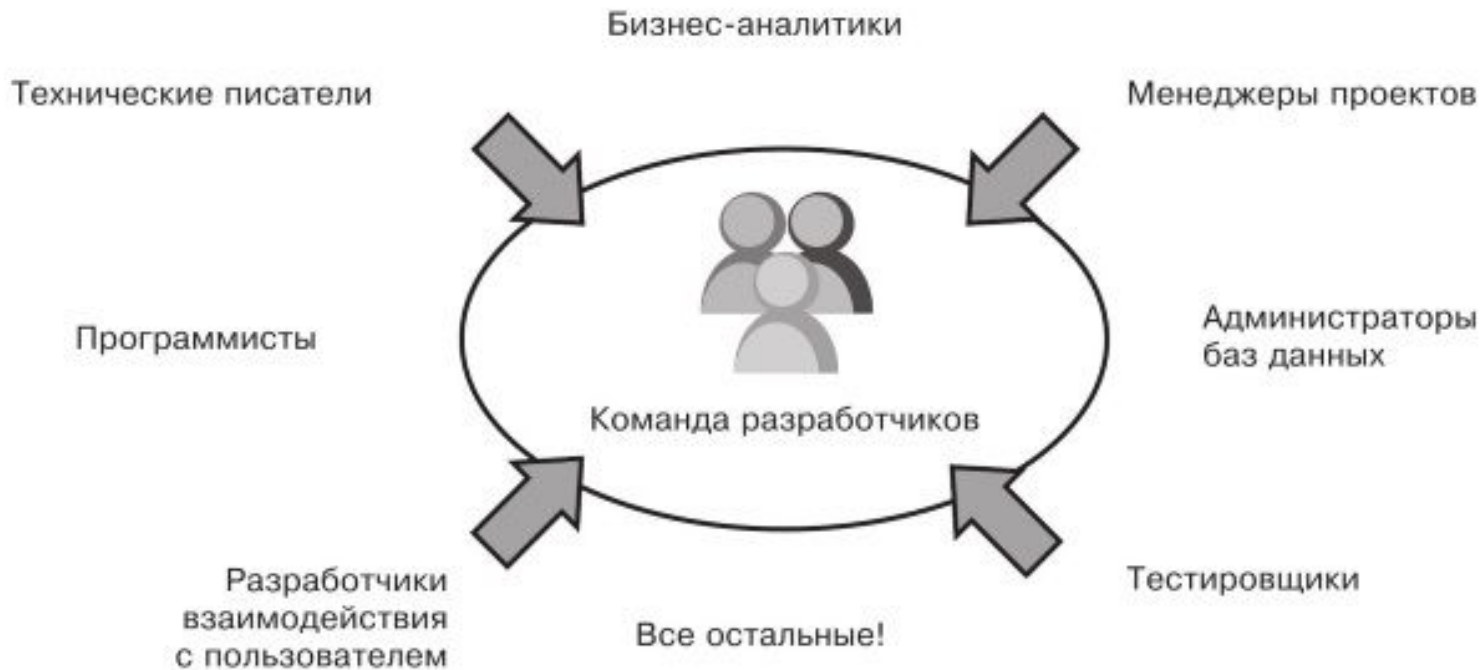
	SUCCESSFUL	CHALLENGED	FAILED
Grand	2%	7%	17%
Large	6%	17%	24%
Medium	9%	26%	31%
Moderate	21%	32%	17%
Small	62%	16%	11%
TOTAL	100%	100%	100%

Отличия от других областей производства

- Очень высокая сложность систем
 - <http://www.informationisbeautiful.net/visualizations/million-lines-of-code/>
- Меньше накоплено опыта
 - Более непредсказуем результат
 - Хуже поддается планированию
 - Больше творчество, чем ремесло
- Подверженность постоянным изменениям

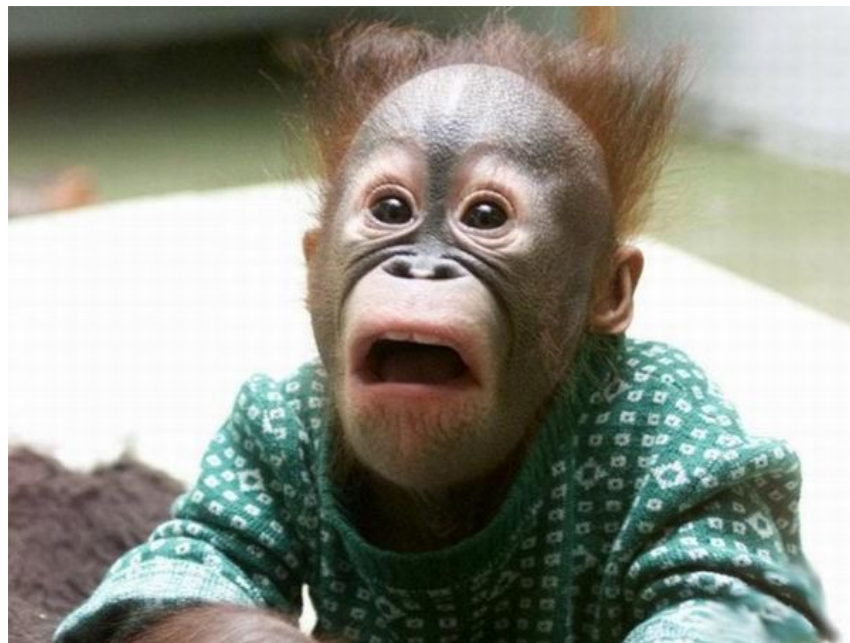


Проектная команда



Программные проекты крайне социализированы

- разрабатываются людьми
 - общение внутри команды
- разрабатываются для людей
 - общение за пределами команды
- успех определяется социальными факторами
 - технологии вторичны?



Профессиональные стандарты

- Стандарт АПКИТ 2009
 - www.apkit.ru/files/programer.doc
- Стандарт АПКИТ 2013
 - http://www.apkit.ru/committees/education/projects/06.001_Programmer.zip

Востребованные компетенции

- умение работать в команде
- владение современными стратегиями и технологиями организации коллективной разработки программного обеспечения, включая системы управления версиями, процессы непрерывной интеграции, стандарты оформления кода и методы инспекции кода
- понимание основных направлений развития методов коллективной разработки, их отличий и целесообразности применения в зависимости от типа решаемых задач и требований организации

Level 1: стажёр, младший разработчик

- Участие в анализе требований и создании сценариев использования продукта
- Участие в разработке различных типов требований к программному продукту
- Разработка кода программного продукта на основе готовых спецификаций на уровне модулей
- Отладка и тестирование кода на уровне модулей
- Участие в интеграции программных компонент в единое целое
- Анализ и оптимизация кода с использованием инструментальных средств для повышения качества изделий и производительности разработки
- Разработка тестовых наборов и тестовых процедур
- Разработка и ведение проектной и технической документации по порученным задачам
- Участие в ревьюировании технических документов
- Участие в измерении характеристик программного проекта

Level 2: программист/разработчик

- Сбор и анализ требований, создание сценариев использования продукта
- Разработка различных типов требований к программному продукту
- Восстановление требований по коду в процессе реинжиниринга
- Разработка детальной технической спецификации на основе высокоуровневых спецификаций по полученным требованиям
- Формализация и контроль корректности требований и/или спецификаций, сформулированных на неформальном языке
- Разработка кода программного продукта на основе готовых спецификаций
- Отладка кода на уровне модулей, межмодульных взаимодействий и взаимодействий с окружением
- Разработка и отладка распределенных и многопоточных приложений
- Интеграция программных компонент

Level 2: программист/разработчик (2)

- Анализ и оптимизация кода с использованием инструментальных средств для повышения качества продуктов и производительности разработки
- Планирование тестирования и разработка тестовых наборов и процедур
- Разработка и адаптация к проекту средств автоматизации тестирования
- Разработка и ведение проектной и технической документации по порученным задачам
- Ревьюирование технических документов
- Измерение характеристик программного проекта
- Анализ эффективности инструментальных средств для проекта
- Инспекция программного обеспечения
- Обучение и консультирование персонала

Level 3: старший разработчик

- Разработка концепции реализации требований и критериев ее успешности
- Разработка технической спецификации по полученным требованиям
- Формализация и контроль корректности требований и спецификаций, сформулированных на неформальном языке
- Планирование выполнения и процесса измерения проекта
- Разработка и ведение проектной и технической документации
- Сдача документации и программного обеспечения заказчику
- Управление выполнением проекта
- Руководство проектной группой
- Взаимодействие с заказчиками
- Участие в совершенствовании процесса разработки в рабочих группах и технических советах
- Участие в выработке корпоративных и проектных стандартов разработки

Level 4: ведущий программист

- Контроль разработки различных типов требований к программному продукту
- Контроль разработки технической спецификации по полученным требованиям
- Контроль разработки кода программного продукта на основе готовых спецификаций
- Контроль интеграции программных компонент
- Контроль разработки и ведения проектной и технической документации
- Анализ результатов выполнения проекта на основе метрик

План курса

- Жизненный цикл ПО, модели ЖЦ. Основные методологии и модели разработки.
- Требования: сбор, анализ, документирование, работа с изменениями. Планирование и управление рисками.
- Управление проектами: зачем нужен менеджер проекта, процессы и инструментальные средства (баг/таск-трекер, CI/CD).
- Проектирование ПО. Архитектура.
- Парадигмы программирования, их применимость.
- Разработка: что такое хороший код.
- Пользовательские интерфейсы: проектирование, анализ, тестирование.
- Команда. Групповая разработка, управление версиями.
- Противодействие ошибкам. Отладка: виды, подходы. Защитное программирование
- Тестирование и QA
- Выпуск, сопровождение и поддержка: жизнь после релиза и product management
- Качество ПО: характеристики, метрики, стандарты

Software Engineering Book of Knowledge

1. Software Requirements
2. Software Design
3. Software Construction
4. Software Testing
5. Software Maintenance
6. Software Configuration Management
7. Software Engineering Management
8. Software Engineering Process
9. Software Engineering Models and Methods
10. Software Quality
11. Software Engineering Professional Practice
12. Software Engineering Economics

Тут живут презентации

- <https://tinyurl.com/se-course>