

О разработке программных продуктов

Юрий Литвинов
y.litvinov@spbu.ru

11.10.2024

Программа и программный продукт



Ф. Брукс, “Мифический человеко-месяц”

Жизненный цикл ПО

- ▶ Последовательность стадий
 - ▶ Состав и последовательность работ
 - ▶ Получаемые результаты
 - ▶ Методы и средства
 - ▶ Роли и ответственности
 - ▶ ...

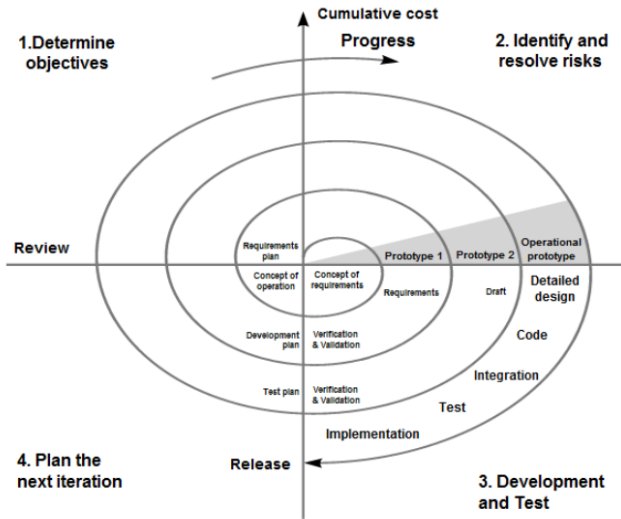
Фазы жизненного цикла ПО

- ▶ возникновение и исследование идеи
- ▶ анализ и сбор требований
 - ▶ пилотный проект
- ▶ планирование и проектирование
- ▶ разработка
- ▶ отладка и тестирование
- ▶ сдача
- ▶ сопровождение
- ▶ вывод из эксплуатации

Водопадная модель



Спиральная модель



Методологии разработки

- ▶ Модели описывают последовательность фаз и что надо делать на этих фазах, методологии — как делать
- ▶ Гибкие методологии
 - ▶ Agile Manifesto
 - ▶ Люди и взаимодействие важнее процессов и инструментов
 - ▶ Работающий продукт важнее исчерпывающей документации
 - ▶ Сотрудничество с заказчиком важнее согласования условий контракта
 - ▶ Готовность к изменениям важнее следования первоначальному плану
 - ▶ SCRUM
 - ▶ XP

Парадигмы программирования

- ▶ Методологии говорят, как делать, но до уровня “а сейчас писать код”, стиль написания кода — выбранная парадигма
- ▶ Структурное программирование
- ▶ Объектно-ориентированное программирование
- ▶ Функциональное программирование
- ▶ Много других

Структурное программирование

- ▶ Дейкстра, 70-е годы XX века
- ▶ Базовые конструкции
 - ▶ Последовательное исполнение
 - ▶ Ветвление
 - ▶ Цикл
- ▶ Произвольная вложенность
- ▶ Никаких других средств управления последовательностью выполнения
- ▶ Выделение подпрограмм
- ▶ Нисходящее проектирование

Нисходящее проектирование

- ▶ Декомпозиция задачи
 - ▶ Подпрограммы
 - ▶ Модули
- ▶ Высокоуровневая реализация с помощью “заглушек”
- ▶ Постепенная реализация модулей и подпрограмм
- ▶ Строгое задание входов и выходов

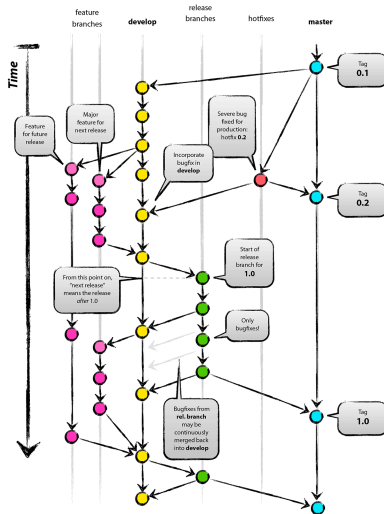
Модули

- ▶ Логически обособленный кусок функциональности
 - ▶ Интерфейс
 - ▶ Реализация
- ▶ Особенности
 - ▶ Четкая декомпозиция
 - ▶ Минимизация
 - ▶ Один модуль — одна функциональность
 - ▶ Отсутствие побочных эффектов
 - ▶ Независимость от реализации других модулей
 - ▶ Принцип сокрытия данных

Практики

- ▶ Документирование
- ▶ Моделирование
- ▶ Тестирование (модульное и регрессионное)
- ▶ Версионный контроль
- ▶ Непрерывная интеграция
- ▶ Code review

Версионный контроль



© <https://nvie.com/posts/a-successful-git-branching-model/>