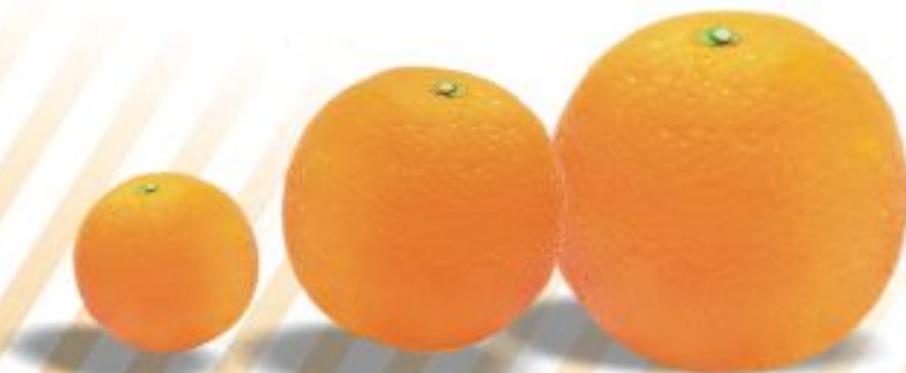


# Тестирования программного обеспечения

**Сержантов Антон,  
Ведущий программист,  
JaNet systems LLC**



**Москва, 2010 г.**



# Тестирование программного обеспечения

- Что такое тестирование, основные понятия.
- Программные ошибки, документирование и анализ
- Разновидности тестирования и их роль в процессе разработки ПО
- Тест-план. Для чего нужен тест-план, какие тесты должен включать?
- Анализ ПО. Приоритеты тестирования.



# Что такое тестирование?

## Основные понятия

### Заблуждения при тестировании:

- Нельзя полностью протестировать продукт
  - Невозможно проверить реакцию программы на каждую комбинацию входных данных
  - Невозможно проверить все способы редактирования данных
  - Невозможно проверить реакцию программы на ввод данных в каждый момент ее работы (окружение)
  - Невозможно проверить каждую возможность выполнения команд программы
  - Невозможно выявить все ошибки проектирования
- Тестирование – проверка правильности программы?
  - Невозможно проверить, что программа работает правильно



# Что такое тестирование?

## Основные понятия

### Цель тестирования:

- Программу тестируют для того, чтобы найти в ней ошибки
- Ошибки ищут для того, чтобы их исправить
- Повышение надежности программы как основной составляющей ее качества
- Повышение качества программы:
  - Качество – точное соответствие программы спецификации клиента?
  - Качество программы определяется возможностями, благодаря которым она понравится пользователю, а также недостатками, которые вынуждают его приобрести другую программу



# Программные ошибки, документирование и анализ

Что является ошибкой в программе:

- Расхождение между программой и спецификацией
- Программа не делает того, чего пользователь от нее вполне обоснованно ожидает



# Программные ошибки, документирование и анализ

## Категории программных ошибок:

- Ошибки пользовательского интерфейса
  - Ошибки функционала
  - Взаимодействие программы с пользователем
  - Организация программы
  - Пропущенные команды
  - Производительность
  - Выходные данные
- Обработка ошибок
- Алгоритмический ошибки (вычисления, потоки, гонки, перегрузки)
- Контроль версий
- Документация

НАЗВАНИЕ КОМПАНИИ _____	КОНФИДЕНЦИАЛЬНО _____	ОТЧЕТ О ПРОБЛЕМЕ № _____
ПРОГРАММА _____	ВЫПУСК _____	ВЕРСИЯ _____
ТИП ОТЧЕТА (1-6) _____	СТЕПЕНЬ ВАЖНОСТИ (1-3) _____	ПРИЛОЖЕНИЯ (Д/Н) _____
1 - Ошибка кодирования	1 - Фатальная	Если да, какие:
2 - Ошибка проектирования	2 - Серьезная	
3 - Предложение	3 - Незначительная	
4 - Расхождение с документацией		
5 - Взаимодействие с аппаратурой		
6 - Вопрос		
ПРОБЛЕМА _____		
МОЖЕТЕ ЛИ ВЫ ВОСПРОИЗВЕСТИ ПРОБЛЕМНУЮ СИТУАЦИЮ? (Д/Н) _____		
ПОДРОБНОЕ ОПИСАНИЕ ПРОБЛЕМЫ И КАК ЕЕ ВОСПРОИЗВЕСТИ _____		
ПРЕДЛАГАЕМОЕ ИСПРАВЛЕНИЕ (НЕОБЯЗАТЕЛЬНО) _____		
ОТЧЕТ ПРЕДСТАВЛЕН СОТРУДНИКОМ _____		ДАТА __/__/__
<i>СЛЕДУЮЩИЕ ГРАФЫ ПРЕДНАЗНАЧЕНЫ ТОЛЬКО ДЛЯ РАЗРАБОТЧИКОВ</i>		
ФУНКЦИОНАЛЬНАЯ ОБЛАСТЬ _____	ОТВЕТСТВЕННЫЙ _____	
КОММЕНТАРИИ _____		
СОСТОЯНИЕ(1-2) _____		ПРИОРИТЕТ (1-5) _____
1 - Открыто      2 - Закрыто		
РЕЗОЛЮЦИЯ (1-9) _____	ИСПРАВЛЕННАЯ ВЕРСИЯ _____	
1 - Рассматривается	4 - Отложено	7 - Отозвано составителем
2 - Исправлено	5 - Соответствует проекту	8 - Нужна дополнительная информация
3 - Не воспроизводится	6 - Не может быть исправлено	9 - Не согласен с предложением
РАССМОТРЕНО _____	ДАТА __/__/__	
ПРОКОНТРОЛИРОВАНО _____	ДАТА __/__/__	
СЧИТАТЬ ОТЛОЖЕННЫМ (Д/Н) _____		

РИСУНОК 5.1 Форма документа “Отчет о проблеме”



# Программные ошибки, документирование и анализ

Каким должен быть отчет об ошибке:

- Простота
- Понятность
- Воспроизводимость
- Беспристрастность



# Программные ошибки, документирование и анализ

## Анализ воспроизводимой ошибки:

- Выявить все наиболее серьезные последствия проблемы
- Найти простейший и кратчайший путь воспроизведения ошибки
- Найти альтернативные действия, приводящие к такому же результату
- Выявить связанные проблемы



# Разновидности тестирования и их роль в процессе разработки ПО

## Этапы разработки ПО:

- Планирование
  - Анализ требований (3%)
  - Спецификация (3%)
- Проектирование (5%)
- Кодирование и написание документации (7%)
- Тестирование и исправление ошибок (15%)
- Поддержка и сопровождение (67%)

Чем раньше найти и исправить ошибку, тем дешевле это обойдется!



# Разновидности тестирования и их роль в процессе разработки ПО

## Тестирование на этапе планирования:

- Что тестируется:
  - Идеи
- Кто тестирует:
  - Отдел маркетинга
  - Руководители проекта
  - Аналитики
  - Архитекторы
  - Специалисты по анализу человеческого фактора



# Разновидности тестирования и их роль в процессе разработки ПО

## Тестирование на этапе планирования:

- Основные вопросы:
  - Адекватны ли требования?
  - Полны ли они?
  - Совместимы ли требования между собой?
  - Выполнимы ли?
  - Разумны ли они?
- Сравнительный анализ существующих продуктов
- Дискуссионные группы



# Разновидности тестирования и их роль в процессе разработки ПО

## Тестирование на этапе проектирования:

### – Что тестируется:

- Идеи, более формализованы
- Проектная документация

### – Кто тестирует:

- Аналитики
- Архитекторы (в основном решение проблем)



# Разновидности тестирования и их роль в процессе разработки ПО

## Тестирование на этапе проектирования:

### – Основные вопросы:

- Действительно ли проект хорош?
- Соответствует ли проект требованиям?
- Полон ли проект?
- Достаточно ли он реалистичен?
- Хорошо ли описана (продумана, спроектирована) в проекте подсистема обработки ошибок?!

### – Совещание аналитиков:

- Обзорное совещание
- Инспекционное совещание
- Рецензионное совещание



# Разновидности тестирования и их роль в процессе разработки ПО

## Разновидности тестирования:

- Тестирование «Стеклянного ящика»:
  - Преимущества:
    - Направленность тестирования
    - Полный охват кода
    - Управление потоком
    - Отслеживание целостности данных
  - Структурное тестирование:
    - Тестирование программных путей, критерии охвата
  - Модульное тестирование, нисходящее и восходящее тестирование
  - Статическое тестирование и динамическое
  - Псевдоотладка и мутационное тестирование
  - Анализ производительности
  - Регрессионное тестирование



# Разновидности тестирования и их роль в процессе разработки ПО

## Разновидности тестирования:

- Тестирование «Черного ящика»:
  - Приемочное тестирование
  - Функциональное тестирование:
    - Сверка со спецификацией
    - Граничные условия
    - Производительность
    - Переходы между режимами
    - Эксплуатация в реальном режиме
    - Нагрузочное тестирование (максимальной объем входных данных, многозадачность, многопользовательский режим)
    - Обработка ошибок
    - Защита
    - Совместимость и преобразование форматов
    - Эффективные тесты



# Разновидности тестирования и их роль в процессе разработки ПО

## Разновидности тестирования:

- Тестирование «Черного ящика»:
  - Системное тестирование:
    - Аппаратные конфигурации
      - Тип ОС
      - Формат дисков
      - Клавиатура
      - Установка
      - Интерфейс
    - Установка и обслуживание



# Тест-план. Для чего нужен тест-план, какие тесты должен включать?

## Для чего нужен тест-план?

- Организация тестирования
  - Обеспечить полноту охвата продукта
  - Избежать лишних повторений и не забыть ничего важного
  - Повышение эффективности тестирования (сходные тесты, сокращение количества тестов без потери охвата тестирования)
- Организация сотрудников
  - Совместное обдумывание стратегии тестирования
  - Обсуждение объема тестирования
  - Обсуждение глубины тестирования и календарного плана работ



# Тест-план. Для чего нужен тест-план, какие тесты должен включать?

## Для чего нужен тест-план?

- Представляет собой удобную структуру для планирования и управления тестирование
  - Оценка ресурсов
  - Организация
  - Координирование
  - Выявление недостатков тест-плана



# Анализ ПО.

## Приоритеты тестирования.

- Более тщательным образом должны быть протестированы те части и документы ПО, с которыми пользователь будет иметь дело в первую очередь!
  - Дистрибутив
  - Первый запуск, начало работы, Tutorials, документация, краткий обзор, видео-презентация, примеры использования
  - Интерфейс, внешний дизайн
    - «Кого будет интересовать, что программный код безупречен, если какая-то часть интерфейса вызывает у пользователей затруднения, путает их, ведет к ошибкам, раздражает или является недостаточно гибкой и функциональной – не делает того, что, по мнению пользователей, она обязательно должна делать».
  - Основной функционал, система обработок ошибок, обновление
  - Функционал



**ООО «ДжаНет системс», ИНН 7709715178**  
**101000, Москва, Петроверигский пер., д. 6-8-10, стр. 8**  
**Тел.: +7 (495) 625-84-00, [info@janetsys.com](mailto:info@janetsys.com)**  
**[www.janetsys.com](http://www.janetsys.com)**