1. Introduction to software testing

Why do we need to test

Astea Solutions QA Team

So, Why do we need to test?

And polite answer...

All software have bugs

There are three types of software:

- Software that have bugs

There are three types of software:

- Software that have bugs
- Software that have less bugs

There are three types of software:

- Software that have bugs
- Software that have less bugs
- Microsoft software

So, Why do we need to test

And naughty answer...

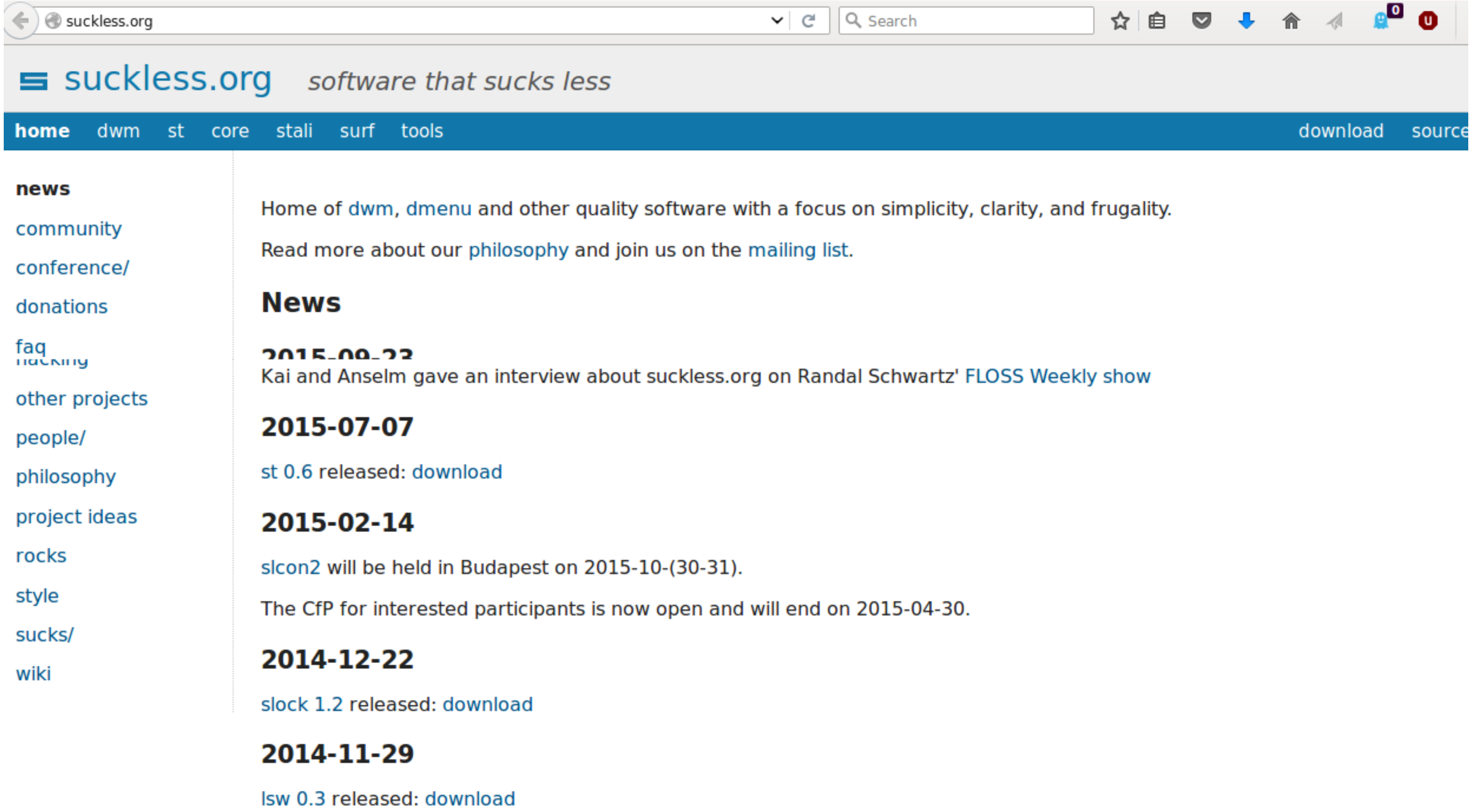
All software sucks

There are three types of software:

- Software that sucks

There are three types of software:

- Software that sucks
- Software that sucks less



suckless.org *software that sucks less*

home dwm st core stali surf tools download source

news
community
conference/
donations
faq
hacking
other projects
people/
philosophy
project ideas
rocks
style
sucks/
wiki

Home of dwm, dmenu and other quality software with a focus on simplicity, clarity, and frugality.

Read more about our philosophy and join us on the mailing list.

News

2015-09-23

Kai and Anselm gave an interview about suckless.org on Randal Schwartz' FLOSS Weekly show

2015-07-07

st 0.6 released: download

2015-02-14

slcon2 will be held in Budapest on 2015-10-(30-31).

The CfP for interested participants is now open and will end on 2015-04-30.

2014-12-22

slock 1.2 released: download

2014-11-29

lsw 0.3 released: download

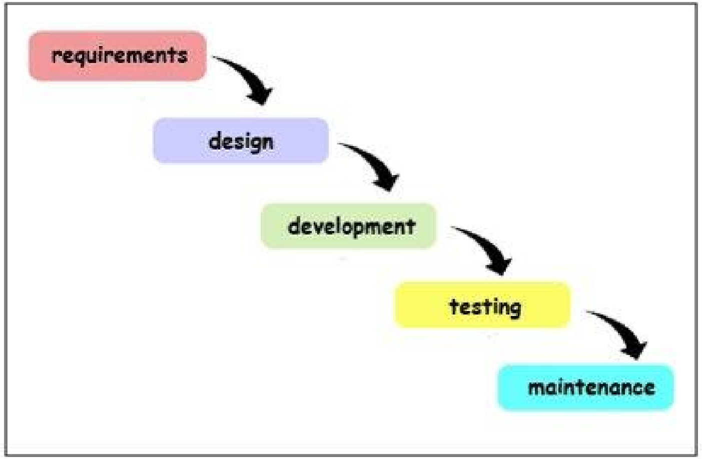
There are three types of software:

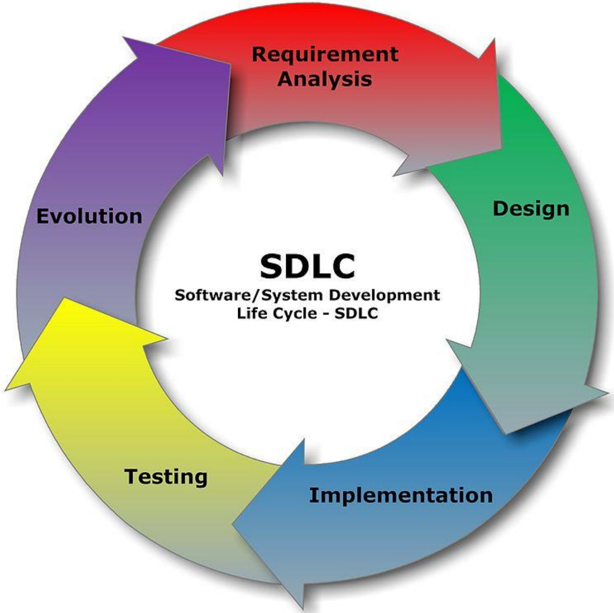
- Software that sucks
- Software that sucks less
- Microsoft software

Overview

- Software Development Life Cycle (SDLC)
- Specification
- Error, defect, and bug terminology
- Testing & Quality
- Why do we test
- Testing principles
- Misconceptions about testing

Software Development Life Cycle (SDLC)





Specification(1)

- Document
- Specifies requirements
- Functional, performance, design requirements; interface descriptions, development standards
- Specification can have bugs
- Lack of specification

Specification (2)

Different types of documents

- Use cases
  - *Actor does something -> System does something -> Actor does something else -> System does something else*
- User Stories
  - *As an internet banking customer I want to list my account balances so that I can understand my financial position.*
  - *As an internet banking customer I want to list transactions on an account so that I can check the details*

- Technical Design, mock-ups, wire-frames

Error, defect, and bug terminology (1)

- Error
  - *A human action that produces an incorrect result.*
- Fault = Defect = Bug
  - *Is present from the time the software was developed or changed yet materialize only when the software is executed, becoming visible as a failure.*
- Failure = Problem = Issue = Incident
  - *A failure is caused by a fault in the software.*
- Defect masking
  - *Fault is hidden by one or more other faults in other parts of the program*

Error, defect, and bug terminology (2)

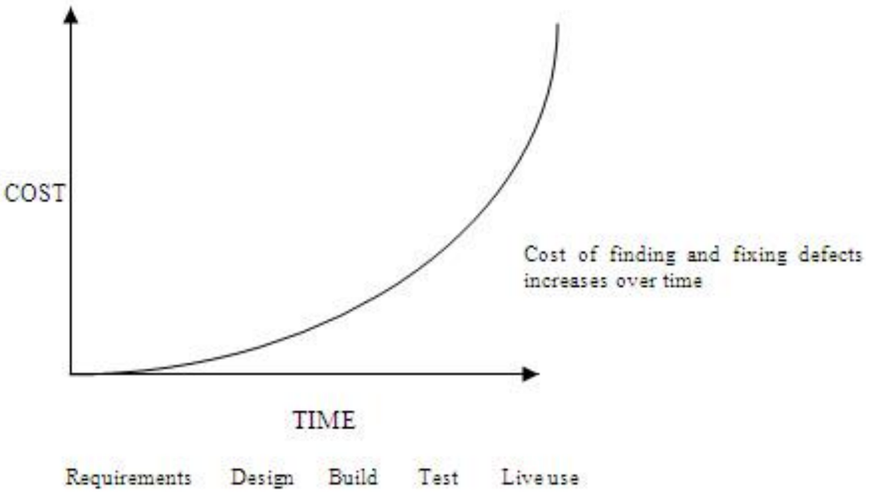
- Cause of errors
  - No one is perfect and we all make mistakes
  - Poor communication
  - Unclear and missing documentation
  - Requirements change
  - Assumptions
  - Time pressure
  - Using new technology
  - ….

Error, defect, and bug terminology (3)

- 3 conditions of a bug
  - Expected result
  - Actual result
  - Expected result ≠ Actual result
- Expected result
  - Specification
  - Experience
  - Common sense - result of logical thinking
  - Standards
  - Statistics
  - Valuable opinion – domain experts

Testing & Quality (1)

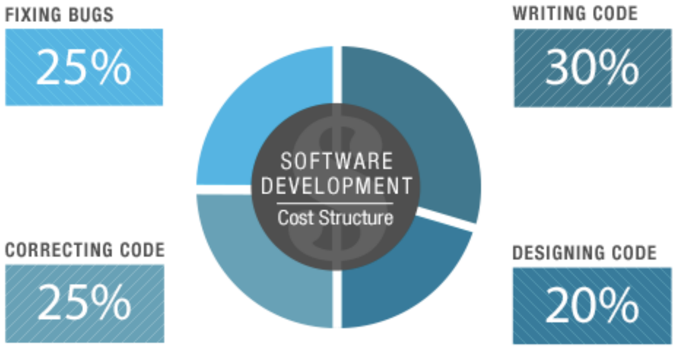
- Cost of errors



Cost of finding and fixing defects increases over time

THE COST OF SOFTWARE BUGS

According to a report by the University of Cambridge, programmers spend nearly half their time correcting code and fixing bugs. The report estimates software bugs cost nearly $312 billion a year.



Testing SHOULD start as early as possible!

Testing & Quality (2)

- Testing
  - satisfy specified requirements
  - demonstrate fit for purpose
  - to detect defects.
- Testing is not debugging
  - Debugging is the task of localizing and correcting faults. Done by developers

Testing & Quality (3)

- Software testing contributes to improvement of software quality
- Software quality comprises the following factors:
  - functionality
  - reliability
  - usability
  - efficiency
  - maintainability
  - portability

Why do we test

- No software developer is perfect

Why do we test

- No software developer is perfect
  - except Linus Torvalds
  - *“My name is Linus Torvalds and I am your god.”*
  - *“You’re a complete incomplete idiot, and I’m not going to apply this patch because it’s obviously broken and is a total piece of sh*t…..And hey, maybe I’m just being a d*ck, and you can prove me wrong, so please explain to me why you did that horrible thing. Please? Hmm?*

Why do we test

- Software development is iterative process
- Find and address bugs before users
- Prove product conforms to specification
- Reduce live defects
    - Increase reliability
    - Increase profitability
    - Keep company reputation
- Prevent unwanted behaviour
- Do our jobs

Testing principles (1)

- Principle 1: Testing shows the presence of defects, not their absence
- Principle 2: Exhaustive testing is impossible
- Principle 3: Testing activities should start as early as possible
- Principle 4: Defect clustering

Testing principles (2)

- Principle 5: The pesticide paradox. Same tests stop finding new bugs
- Principle 6: Testing is context dependent
- Principle 7: “No failures means the system is useful” is a fallacy

Misconceptions about testing (1)

- Testing is too expensive
- Testing is time-consuming
- Only fully developed products are tested
- Complete testing is possible
- A tested software is bug-free
- Number of bugs before release matters

Misconceptions about testing (2)

- Missed defects are due to testers
- Test automation solves every problem
- Testers are responsible for quality of product
- Testing is not technically challenging / is boring
- We can go without testing

QUESTIONS

Summary

- Define the terms failure, fault, and error
- What is defect masking?
- Explain the difference between testing and debugging
- List several misconceptions about testing
- Why do we test

Resources

1. ISTQB Foundation level syllabus, chapter 1 - Fundamentals of testing

Homework

Let’s have a history lesson :)

Make a research and list 3 critical bug stories.

Write down in a document a few sentences with description of the story. Submit your work in Moodle.