Non-functional testing

IT Learning &
Outsourcing Center

Lector: Viktor Slavchev

E-mail: mr.slavchev@gmail.com
Twitter:@TheTestingTroll
Blog: mrslavchev.com

www.pragmatic.bg

Functional vs. non-functional

Can:

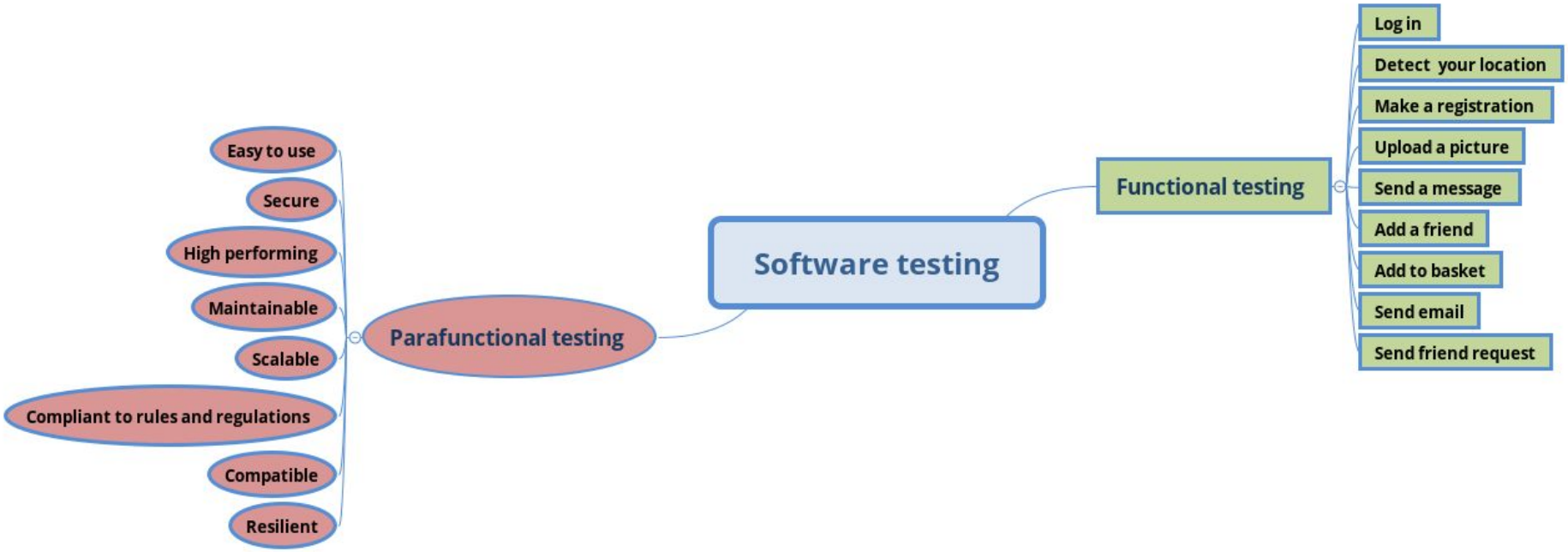
- Make calls
- Make pictures
- Use as messenger
- Use for GPS location
- Play games
- Send/receive mails
- etc

Does certain functions

- Other characteristics:
- Design
- Physical characteristics
- Aesthetics
- Security
- Endurance

Has some properties

Mindmap


Software testing
Functional testing
Log in
Detect your location
Make a registration
Upload a picture
Send a message
Add a friend
Add to basket
Send email
Send friend request
Parafunctional testing
Easy to use
Secure
High performing
Maintainable
Scalable
Compliant to rules and regulations
Compatible
Resilient

Functional vs. non-functional

- **Functional testing** - Testing that aims to expose the risks in software's behaviour.
- Think of functionality as **"what can software do"**.
- Or an action that the software can do.
- **Ex.** Can a user log in, can he create an account, can he make a payment, can he cancel a transaction.
- Often directed by functional requirements.
- Functional coverage to monitor and measure

Functional vs. non-functional

- **Non-functional testing** - should be thought as a set of properties that the software has.
- Non-functional characteristics are often addressed as ...abilities.
- So, another way to think of them might be - **ability to be** …
- Ex. ability to be usable, ability to be maintainable, ability to be secure, ability to perform up to a specific benchmark etc.

Types of non-functional testing

- Usability testing
- Performance/load/stress testing
- Installability/Uninstallability
- Maintainability
- Scalability
- Portability
- Localization and internationalization testing.
- Compliance.
- Recovery testing
- Security and penetration testing.

Non-functional or parafuncitonal

- Problems:
  - It is not entirely non-functional, as some of the types are tested using functions (performance, load, stress testing)
  - Types of NFT are way too different to be described all as non-functional. Ex. security testing and usability testing.
- I prefer the term parafunctional testing: *"focuses on those aspects of the product that aren't tied to specific functions"* (Kaner)
- Functional or non-functional by Rich Rogers

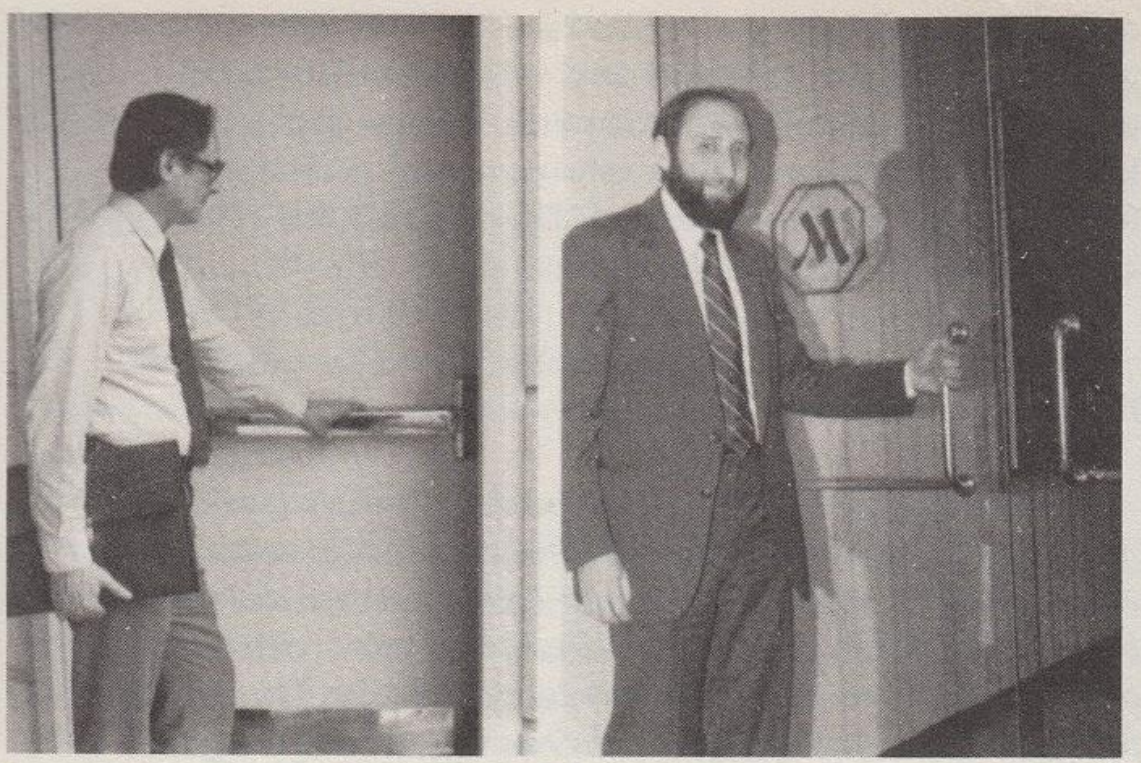
Functional vs. parafunctional

- "**Functional testing** focuses on capability of the product and benefits to the end user. The relationships that are especially valuable to foster are between testers and customers (or customer advocates), so that testers can develop deeper understanding of the ways to prove that this product will not satisfy the customer's needs"
- "**Parafunctional testing** focuses on those aspects of the product that aren't tied to specific functions….Unlike functional aspects of the product, which users and customers understand and relate to pretty well, users and customers are not experts in the parafunctional aspects of the product. Effective testing of these attributes will often require the testers to collaborate with other technical staff (such as programmers)."
  Kaner "The ongoing revolution in testing"

Usability testing.

- Usability testing is the ability of the software product to be easy and understandable to use by its **users.**
- Usability has its roots deep into industry, not just software development: The Design of Everyday Things
- The users are key element in usability testing, expert testers might guide and monitor it, but aren't always the best source of feedback.
- Good source: Software Quality Characteristics

Door handles and usability



1.5 **Affordances of Doors.** Door hardware can signal whether to push or pull without signs. The flat horizontal bar of *A* (above left) affords no operations except pushing: it is excellent hardware for a door that must be pushed to be opened. The door in *B* (above right) has a different kind of bar on each side, one relatively small and vertical to signify a pull, the other relatively large and horizontal to signify a push. Both bars support the affordance of grasping: size and position specify whether the grasp is used to push or pull—though ambiguously.

Why is usability important?

We have to meet clients expectations

Goals of usability testing.

- **Mnemonic - EARE**
- **Efficiency** - How much time, and how many steps, are required for people to complete basic tasks?
- **Accuracy** - How many mistakes did people make? (And were they fatal or recoverable with the right information?)
- **Recall** - How much does the person remember afterwards or after periods of non-use?
- **Emotional response** - How does the person feel about the tasks completed?
  Is the person confident, stressed? Would the user recommend this system to a friend?

Usability testing.

Methods.

- **Hallway testing** - users picked randomly from the hallway and asked to perform specific tests on the software
- **A / B testing** - comparison between 2 or more designs offering different approach to a certain design or issue.
- **Expert review** - usability testing being performed by testing professional.
- **Remote usability testing -** often data-driven approach of gathering information about user and how they interact with software - most used paths, most clicked areas, most visited pages. View more here.

Heuristic evaluation.

- A **heuristic evaluation** is a usability inspection method for computer software that helps to identify usability problems in the user interface (UI) design. It specifically involves evaluators examining the interface and judging its compliance with recognized usability principles (the "heuristics").
- Nielsen's heuristics.
- Sample usability checklist

Nielsen's heuristics

- Usability heuristics:
  - Visibility of system status
  - Match between system and the real world
  - User control and freedom
  - Consistency and standards
  - Error prevention
  - Recognition rather than recall
  - Flexibility and efficiency of use (accelerators)
  - Aesthetic and minimalist design
  - Help users recognize, diagnose, and recover from errors
  - Help and documentation

A Software Tester's Guide To Usability

Rules of usability.

- Some examples for terrible UX:
  - https://www.flickr.com/photos/26409938@N00/4645386550/in/photostream/
  - https://www.lingscars.com/
  - https://gfycat.com/ParchedUnconsciousFlyingfox
  - Demo of real usability problem: https://www.kinoarena.com/bg/

Compatibility testing

- Tests the ability of software to be consistent with its environment or future/former version of itself.
- Types:
  - **Hardware compatibility.**
  - **OS compatibility**.
  - **Compatibility with specific technology (MySQL, Oracle)**
  - **Browser compatibility.**
  - https://codepen.io/MartijnCuppens/pen/MXojmw

L10n, i14n testing

- Testing the system against various locales and nationalities, when multinational software is produced. Broken cyrillic subtitles
- Language:
  - Testing with standard and extended latin charset: ex. á, é, í, ó, ú, ü, ñ, ¿, ¡
  - Test with languages with right-to-left orientation(arabic) or up down orientation (chinese)
- Currency: conversions, keep in mind ones that have symbol in front - $100 and where the symbol is trailing - 100 €

L10n, i14n testing

- Phone numbers: including special symbols
- Floating point separator - ‘.’ or ‘,’
- Dates: yyyy/mm/dd or dd/mm/yyyy
- Hour format - 24h or 12h A.M./P.M.
- Postal codes: 1000, GY1 123
- VAT numbers BG999999999 or IE9700053D
- Different time zone
- Sorting priorities based on specific symbol system.
- Keyboard usage
- Legal requirements ex. Preselected Terms.
- Good source for any input test: List of naughty strings

Security and penetration testing.

- What could be a security risk?
  - Information in general.
  - Sensitive data hardcoded in the source of errors.
  - Misconfigurations.
  - Custom frameworks, that lack basic security measures.
  - Poor management to virtual/physical access rights.
  - Disclosure of important data.
  - Loss of important data.
  - Risk for your business or the business of your clients.

Security and penetration testing.

- Threat modeling - video by Dan Billing.
  - Infrastructure.
  - Development framework.
  - Databases.
  - Entry points.
  - Assets.
  - Interactions of trust.

Penetration testing.

- Aims to expose system security vulnerabilities, from the perspective of an ethical hacker.
- Has 5 phases:
  - **Reconnaissance** - gathering information about your target via a medium or by yourself.
  - **Scanning** - more disclosure - port scanners, vulnerability scanners.
  - **Gain access** - actually performing the task/hacking.
  - **Maintaining the access**.
  - **Covering our tracks**/writing your reports.

Security and penetration testing.

- Techniques to be used:
  - Spoofing
  - Session hijacking/cookie stealing.
  - Man in the middle attacks.
  - DoS/DDoS - denial of service.
  - Credentials brute forcing.
  - Port sniffing.
  - Packet sniffing.
  - DLL injection
  - SQL injection
  - Cross-site scripting
  - Social engineering - non technological.

SQL injection example

- 1;DROP TABLE users

Gijs in 't Veld
@gintveld

Follow

Great to see that my name still causes SQL errors and that errors thrown are so hacker friendly. ;-) #integrate2016

Invalid query: You have an error in your SQL syntax; check the manual that corresponds to your MariaDB server version for the right syntax to use near 't Veld', NOW()), ('cc89fdd01ea0', 'User-Profile', ':=', 'free750', '', '', NOW(' at line 1 Whole query: INSERT INTO newusers (username, attribute, op, value, callingstationid, displayname, created_at) VALUES ('cc89fdd01ea0', 'Cleartext-Password', ':=', '70358542', 'cc-89-fd-d0-1e-a0', 'Gijs in 't Veld', NOW()),('cc89fdd01ea0', 'User-Profile', ':=', 'free750', '', '', NOW())

But most of all...

A guy named Samy Kamkar made attack against Myspace profiles by submitting XSS which made his victims to send him friend request and add the following description to their profile "but most of all, Samy is my hero".

More info here: https://samy.pl/popular/

Penetration tools.

- THC Hydra - tool for brute forcing web passwords
- fcrackzip - if you want to play with brute forcing tools
- Metasploit - database of exploits for various systems.
- KaliLinux - Linux OS distribution for PenTesters
- Damn vulnerable web application - DVWA - good test env to sharpen your claws.

Security and penetration testing.

- Additional resources:
  - OWASP - open source web security resources
  - Cybrary - free online course in penetration testing
  - Offensive security - pentesting certification
  - Pentest magazine - magazine for pentesting

Questions