

10α Έλεγχος και επαλήθευση λογισμικού

Τεχνολογία Λογισμικού

Σχολή Ηλεκτρολόγων Μηχανικών & Μηχανικών Υπολογιστών
Εθνικό Μετσόβιο Πολυτεχνείο

Χειμερινό εξάμηνο 2017-18

Δρ. Κώστας Σαΐδης (saiko@di.uoa.gr)

Δύο κύρια ερωτήματα

1. Πώς εξασφαλίζουμε ότι το λογισμικό είναι σύμφωνο με τις απαιτήσεις και προσδοκίες των χρηστών (λειτουργικές απαιτήσεις);
2. Πώς διαφυλάσσουμε την ποιότητα του λογισμικού (μη λειτουργικές απαιτήσεις);

Σε πολλούς άξονες:

- Χρηστικότητα / ευχρηστία
- Επεκτασιμότητα / βιωσιμότητα
- Ασφάλεια / ακεραιότητα δεδομένων
- Αποδοτικότητα / αποκρισιμότητα
- κ.λπ.

Έλεγχος, επαλήθευση και επιβεβαίωση λογισμικού

Έλεγχος

- Επιθεωρήσεις (Reviews)
- Δοκιμασίες (Testing)
- Στατική ανάλυση (Static analysis)

Επαλήθευση

Παράγουμε το λογισμικό σωστά;

Επιβεβαίωση

Παράγουμε το σωστό λογισμικό;

Ελαττώματα (defects, bugs)

There are two ways to write error-free programs; only the third one works.

Alan J. Perlis

Ελαττώματα (defects, bugs)

Σοβαρότητα (severity)

- Critical
- Major
- Minor
- Trivial

Κύκλος ζωής δοκιμασιών (test life-cycle)

- Επισκόπηση απαιτήσεων
- Σχεδιασμός δοκιμασιών
- Καθορισμός περιβάλλοντος δοκιμασιών
- Εκτέλεση δοκιμασιών
- Παραγωγή αναφορών

Πλάνο δοκιμασιών (test plan)

- Έγγραφο, επιμέρους ή συνολικό (π.χ. master test plan)
- Τι θα ελεγχθεί και πώς (προσέγγιση)
- Παραδοτέα
- Εκτιμήσεις χρόνου/κόστους
- Προγραμματισμός, προσωπικό, εκπαίδευση, έλεγχοι/αποδοχή, κτλ.

Σενάριο δοκιμής

Test case

- Συνθήκες ή μεταβλητές που καθορίζουν την ορθή συμπεριφορά του χαρακτηριστικού που υπόκειται σε έλεγχο
- Συνήθως γράφεται απευθείας σε κάποιο εργαλείο

Test scripts

- Αυτοματοποιημένοι έλεγχοι (test automation)
- Continuous integration (CI)

Βασικά επίπεδα δοκιμασιών

Από το μικρό στο μεγάλο:

- Μοναδιαία δοκιμασία (unit test)
- Δοκιμασία ολοκλήρωσης (integration test)
- Δοκιμασία συστήματος (system test)
- Δοκιμασία αποδοχής (acceptance test)

Μέθοδοι εκτέλεσης δοκιμασιών

- Black box testing (behavioral testing)
 - Έλεγχος της "εξωτερικής" συμπεριφοράς ενός συστατικού ως συνόλου (π.χ. εκτελώντας πειράματα στο public interface του)
- White box testing (structural testing)
 - Έλεγχος της "εσωτερικής" δομής / συμπεριφοράς ενός συστατικού (π.χ. εκτελώντας πειράματα στην εσωτερική του κατάσταση)
- Gray box testing (συνδυασμός)

Unit testing

Έλεγχος μοναδιαίων ενοτήτων του λογισμικού

- Το ελάχιστο στοιχείο του λογισμικού που μπορεί να ελεγχθεί (π.χ. μέθοδος)
- White box testing (όλες οι μέθοδοι, public, private, κ.λπ)
- Δειγματοληψία των πιθανών εισόδων και δοκιμή τους αν παράγουν αστοχία (σφάλμα, λανθασμένη έξοδο)

Integration testing

Έλεγχος "συνδυασμού" πολλών ενοτήτων

- Διαλειτουργικότητα μεταξύ ενοτήτων (όχι αναγκαστικά μοναδιαίων)
- Είτε white, είτε black, είτε gray box testing



2 UNIT TESTS, 0 INTEGRATION TESTS

System testing

Έλεγχος του συστήματος συνολικά

- Ως ολότητα σε σχέση με το λειτουργικό του περιβάλλον
- Δικτύωση, υπηρεσίες, συστήματα αποθήκευσης, υλικό, κ.λπ.

Acceptance testing

Έλεγχος αποδοχής του λογισμικού

- Customer / user acceptance testing
- Black box testing
- Ad hoc testing

Τύποι δοκιμασιών (ανεξάρτητα επιπέδου)

- Smoke testing
- Functional testing
- Usability testing
- Security testing
- Performance testing
- Regression testing
- Bebugging
- Mutation testing
- κ.ά

Smoke testing

Επίπεδα: Integration, System, Acceptance

- Μήπως πήρε φωτιά;
- Δουλεύει τίποτα ή έχουν αποτύχει όλα;
- Κάλυψη όλων των βασικών λειτουργιών οριζόντια, αλλά όχι σε βάθος (κάθετα)
- Να προχωρήσουμε σε επιμέρους δοκιμασίες ή το σύστημα δεν είναι ώριμο ακόμα (θέλει "κι άλλη δουλειά");

Functional testing

Επίπεδα: Unit, System, Acceptance

- Επαλήθευση συμβατότητας με προδιαγραφές και απαιτήσεις
- Καλό είναι να εκτελούνται από τους "χρήστες"
- Domain-driven design, behavior-driven development
- Automated tests/specifications

Usability testing

Επίπεδα: System, Acceptance

- Ad-hoc testing
- A/B testing (σε ευρεία χρήση στις διαδικτυακές εφαρμογές για τη "μέτρηση" της αποτελεσματικότητας μιας UI/UX επιλογής)

Security testing

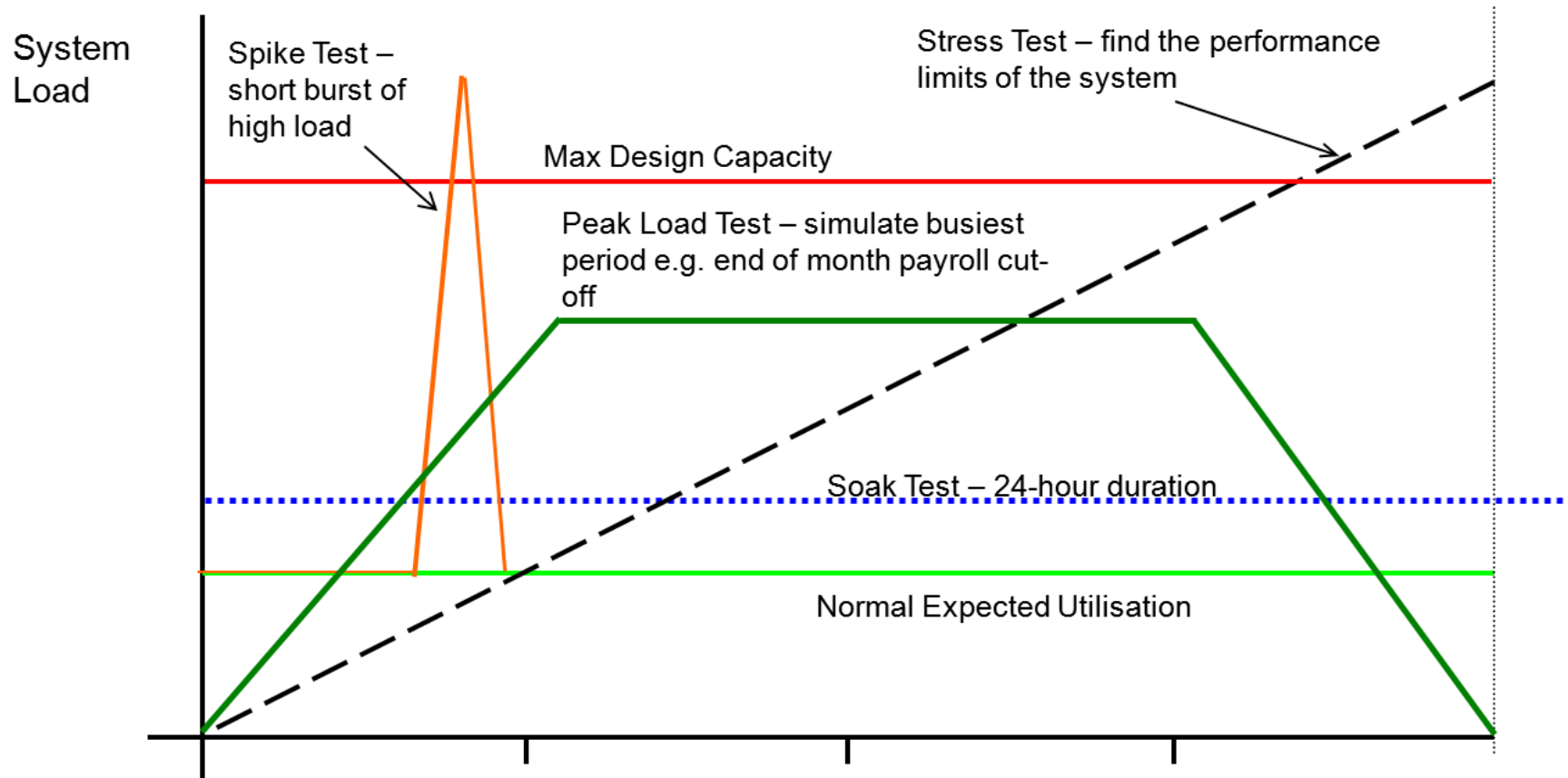
Μια κατηγορία από μόνο του

Για διαδικτυακές εφαρμογές: owasp.org

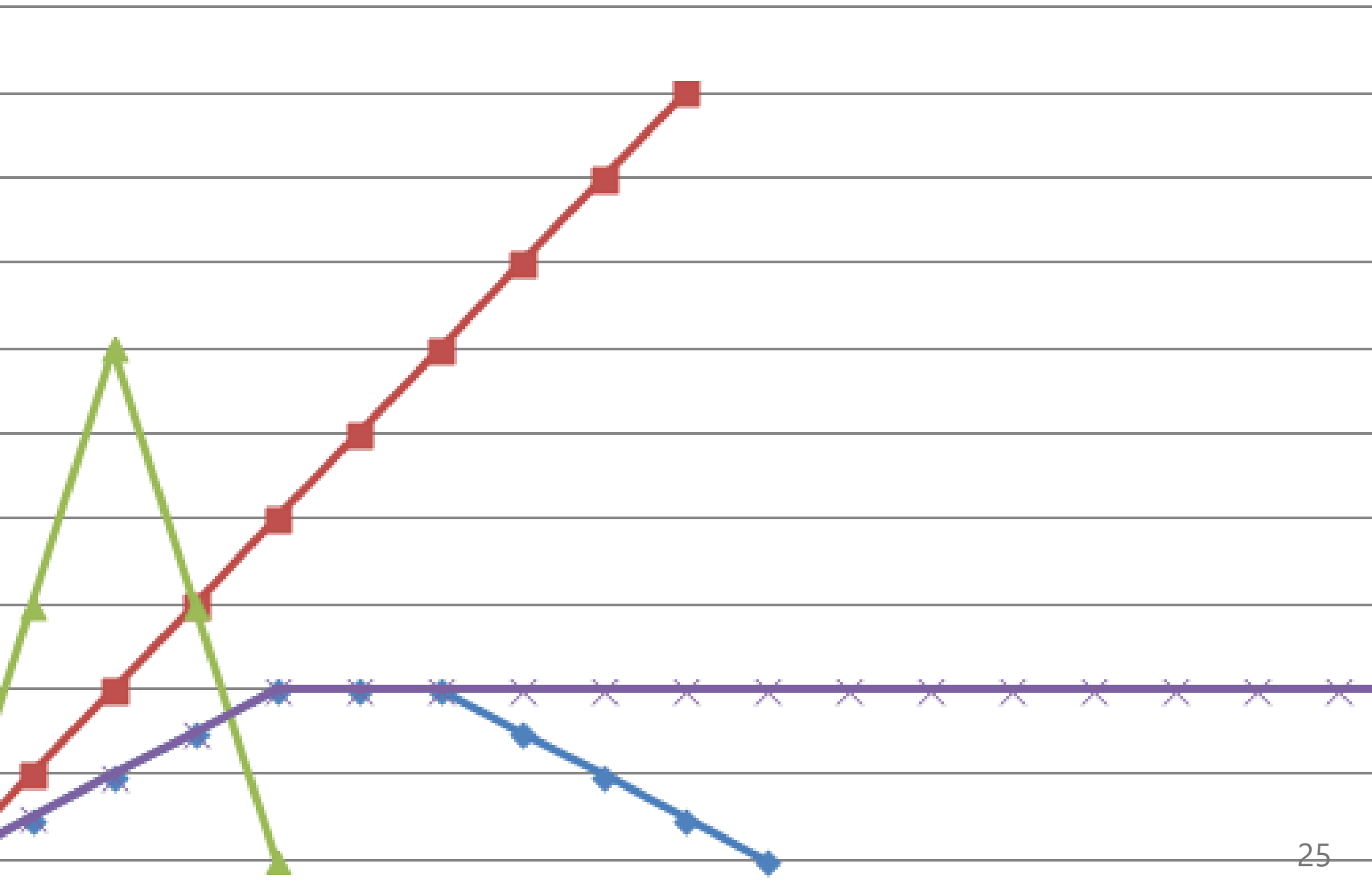
Performance testing

Μια κατηγορία από μόνο του

- Load testing (stress, spike, soak testing)
- Endurance testing, resilience testing, saturation testing, scalability testing
- Maximum usable capacity (concurrent users, throughput)
- Acceptable response time (latency)
- Εργαλείο: [Apache JMeter](#)



◆ Load ■ Stress ▲ Spike ✕ Soak



Regression testing

Οπισθοδρόμηση

- Έλεγχος ότι μια νέα έκδοση του λογισμικού (new feature, bug fix, κτλ.) δεν προκαλεί πρόβλημα στην υπάρχουσα λειτουργικότητα
- Από τους βασικούς ελέγχους σε παραγωγικό λογισμικό

Επίσης

Bebugging

- Εκ προθέσεως εισαγωγή σφαλμάτων στον κώδικα με στόχο να ανακαλυφθούν από τις δοκιμασίες

Mutation testing

- Mutation του κώδικα (π.χ. αλλαγή εκφράσεων σε συντακτικό επίπεδο) με αντίστοιχη λογική

Εργαλεία ελέγχου στο JVM

- JUnit (unit testing)
- TestNG (unit testing)
- Mockito (mockup testing)
- Selenium (web browser automation)
- Jenkins (Continuous integration)
- Spock (specifications)

Σε (επόμενο) φρονιστήριο

Θα δούμε παραδείγματα από:

- Junit
- Spock

