

Calitate și Testare Software (CTS2021)

Prof. univ. dr. Catalin Boja

catalin.boja@ie.ase.ro

<http://acs.ase.ro>



Dep. de Informatică și Cibernetică Economică
ASE București

Organizare

Obiectiv: Însușirea aspectelor practice și teoretice privind conceptele de calitate și testare software

Organizare:

- 14 cursuri și 14 laboratoare în 12 săptămâni
- 2 cursuri și 2 laboratoare sunt programate în afara programului normal (1 seminar pentru test laborator și 1 seminar pentru alte activități)

Evaluare:

- Seminar (pondere nota finală 50%): test practic la laborator (20%), activitate seminar (15%), grile de 5-10 minute la curs (5%), test grila din toată materia în ultima săptămână (10%)
- Examen (pondere nota finală 50%)

Condiții susținere/promovare examen:

- Intrarea în examen (sesiunea normală) este condiționată de acumularea a minim 2 pct din 5
- În examen trebuie obținută nota minimă 5

Structura

- Gestiunea versiunilor codului sursa - Git, SVN
- Principii privind scrierea codului sursă - Clean Code
- Design Patterns
- Unit Testing – Junit
- Concepte privind calitatea software – metrice software, analiză statică a codului (Sonar Qube)

Bibliografie

- Scott Chacon, Bean Straub - *Pro Git, 2nd edition*, Apress, 2014, disponibilă online la adresa <http://git-scm.com/book/en/v2>
- Robert C. Martin - *Clean Code, A Handbook of Agile Software Craftsmanship*, Prentice Hall, 2009
- Erich Gamma, Richard Helm, Ralph Johnson, John Vlissides - *Design Patterns: Elements of Reusable Object-Oriented Software*, Addison-Wesley, 1994 ([link Google books](#))
- Steve Holzner - *Design Patterns for Dummies*, Wiley, 2006
- Lasse Koskela - *Effective Unit Testing*, Manning, 2013
- Lasse Koskela - *Practical TDD and Acceptance TDD for Java Developers*, Manning, 2007
- Alasdair Allan - *Pragmatic Unit Testing in Java with JUnit*, The Pragmatic Programmers, 2004
- <http://git-scm.com/docs>
- <https://try.github.io> at [Code School](#)
- <http://acs.ase.ro/software-quality-testing>

Cunoștințe necesare



- Cursul de Programare în Java (implementare concepte POO în Java)
- Cursul de Programare Orientată Obiect (concepte generale POO, sintaxa C/C++)
- Cursul de *Dezvoltare Aplicații Mobile si Programare Aplicațiilor Windows* (concepte legate de arhitecturi de aplicații)

Cunoștințe necesare



- Dezvoltarea de aplicații Java pentru consolă
- Definire clase, clase abstracte, interfețe
- Moștenire
- Lucrul cu stream-uri (fișiere binare și text)

Instrumente necesare

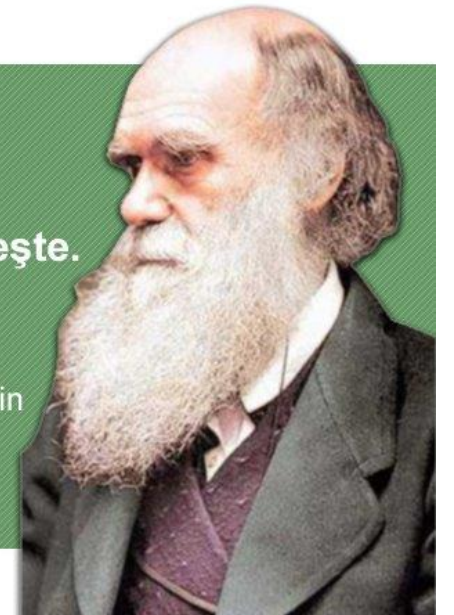


De ce QA si testare ?

"THE ONLY THING
THAT IS CONSTANT
IS CHANGE."
-HERACLITUS

Nu specia cea mai puternică
sau cea mai inteligentă supraviețuiește.
Ci aceea care se adaptează
cel mai bine la schimbări.

Charles Darwin

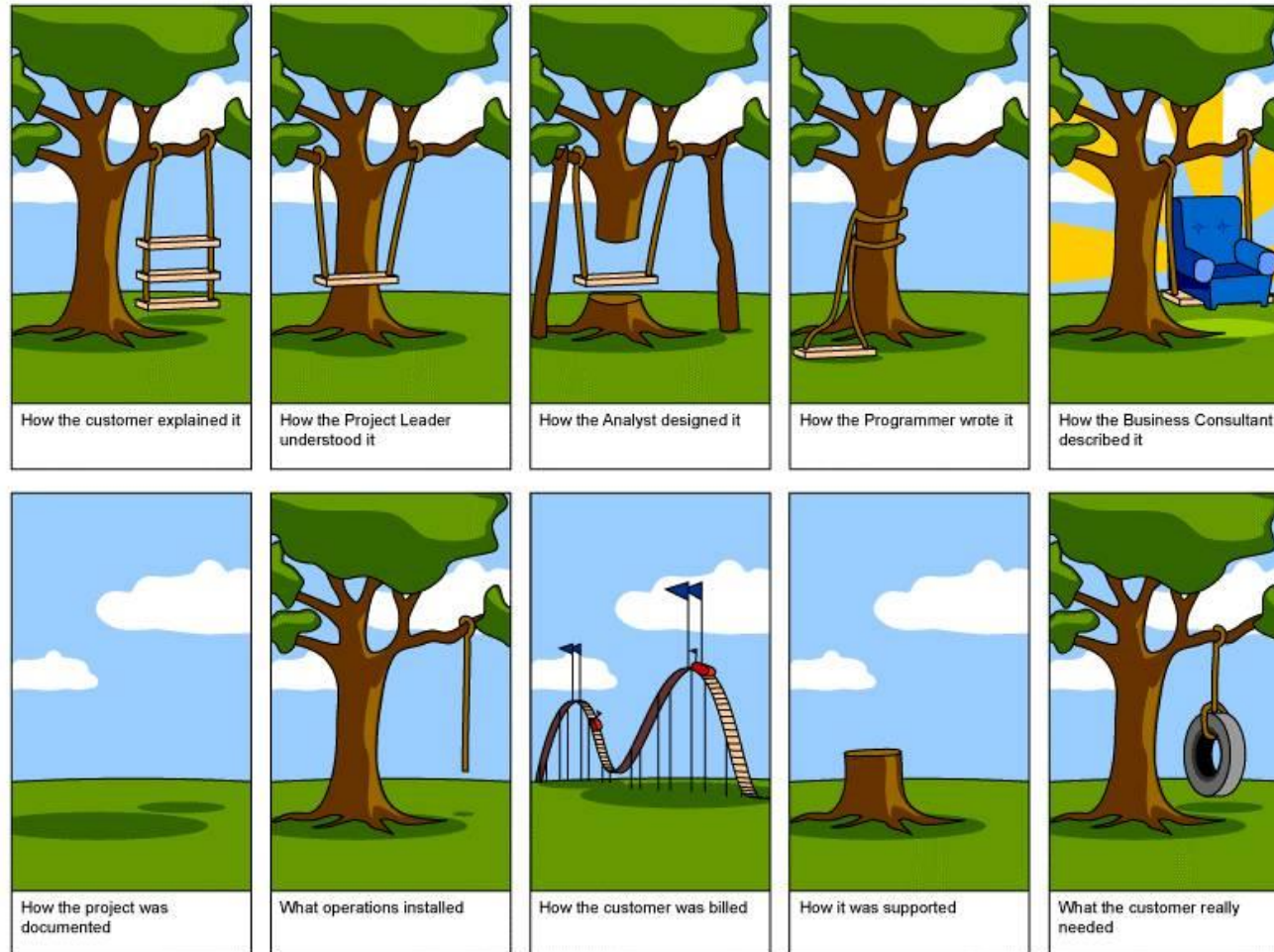


CuvinteCelebre.ro

De ce QA si testare ?

- *Cam la fiecare 5 ani populația de programatori se dublează (începând cu anii 70) [Bob Martin]*
- In 2013 erau estimați 18.2 dezvoltatori software la nivel mondial, iar in 2019 numărul va fi de 26.4 milioane (cu 45% mai mult)
(<http://www.computerworld.com/article/2483690/it-careers/india-to-overtake-u-s--on-number-of-developers-by-2017.html>)
- Foarte mulți programatori nu au suficientă experiență

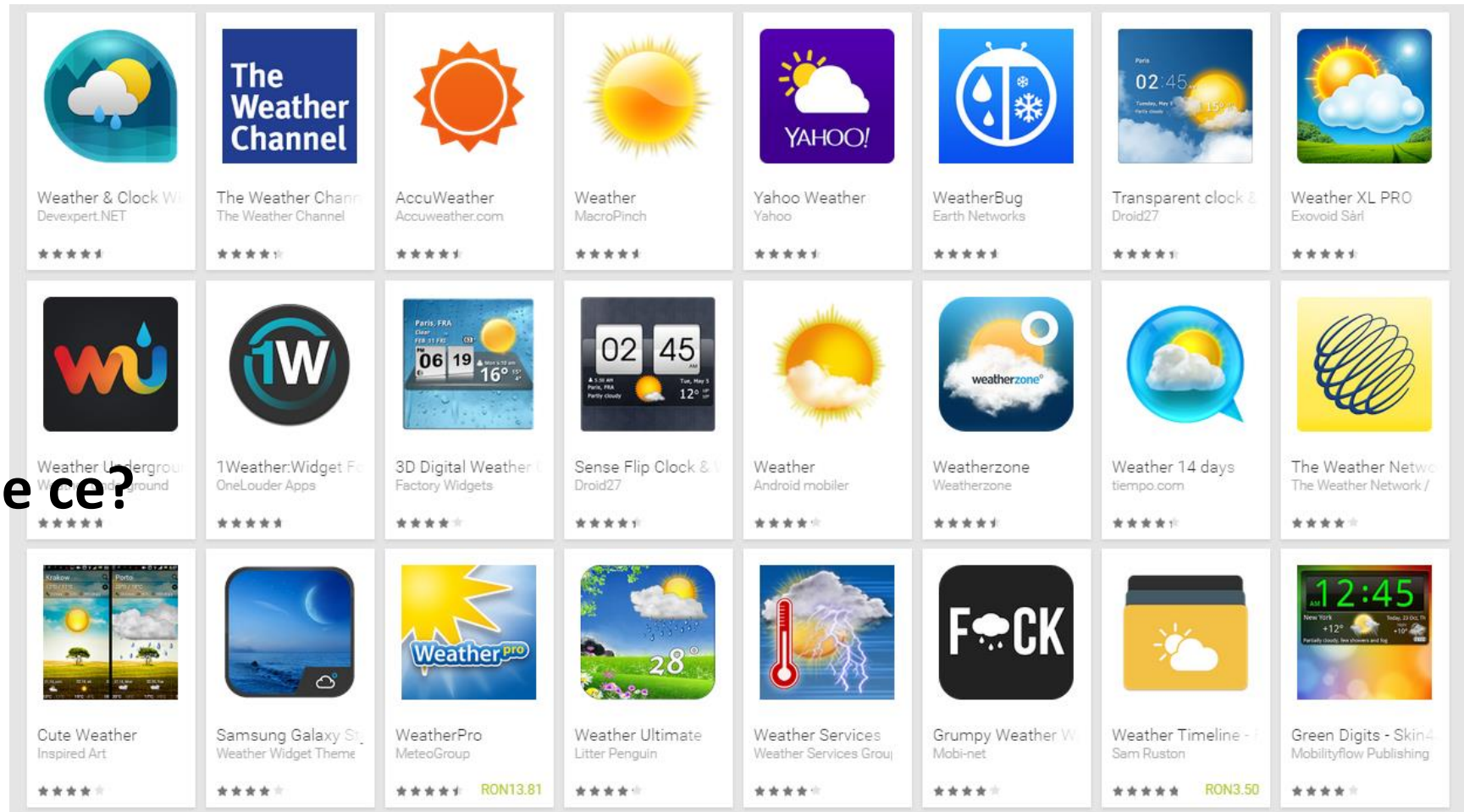
De ce QA si testare ?



2005 Paragon Innovations, Inc.

De ce QA si testare ?

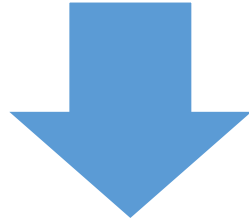
Ce alegi ? și De ce?



QA vs Testare

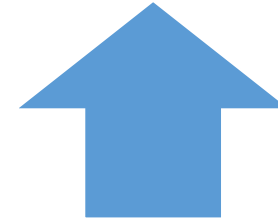
Calitate Software

- Definirea calității produsului
- Îmbunătățirea procesului de dezvoltare
- Prevenirea apariției erorilor și a bug-urilor

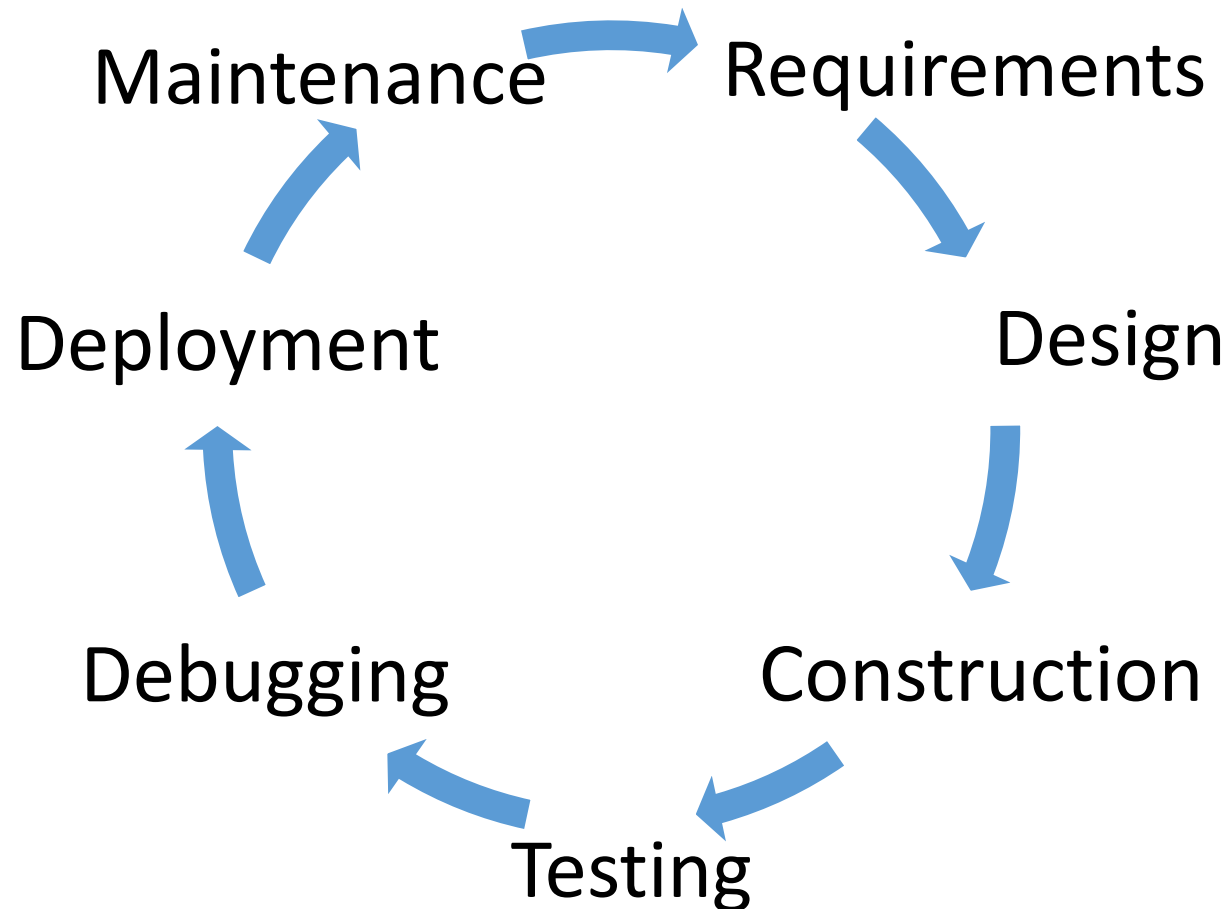


Testare Software

- Identificarea erorilor și a bug-urilor înaintea utilizatorilor



Dezvoltarea software



Rolul responsabilului cu testarea



Viziunea responsabilului cu testarea

