Test education for Universities and Universities of Applied Science, an update after 4 years

Jos van Rooyen

identify
THE QUALITY DRIVER

- Employed at Identify as partner / principal consultant
- 30 years in software testing & quality management
- Co-author TestGrip, TestFrame, Project de Baas, Quality Supervision, Textbook; “Aan de slag met software testen”, Cleantxt, Test Automation Architecture (available soon)
- Test expert online magazine Computable
- Publication areas; Test process Improvement, BI-testing, Test automation, Test Education, Risk Based Testing, Quality Supervision
- Member NESMA working party; Metrics in Contracts
- Visiting lecturer Universities of Applied Science
- Member advisory board Hogeschool Utrecht
- Member of several working parties Dutch Testing Society
  - Member of the board
  - Test Education universities of applied science

Interpretation of the problem (1)

- There were several reasons to start working on this topic;
  - Companies depend more and more to IT
  - Software glitches still increase
  - The complexity of the IT is increasing. Test must increase the quality level to stay in line
  - New development methods, like Agile, requires other skills
  - Test must become more professional

Interpretation of the problem (2)

Software quality and testing was not really embedded in the curricula of universities of applied sciences

Scope of the working party

The working party has (originally) defined 4 goals:

- Inventory of the current situation in the Netherlands
- Is there a need for a new study?
- Defining a curriculum for the new study
- Implementation of the new study?

Scope of the working party (2)

During the second phase the scope was expanded:

- Defining a minor / specialization on software testing

History

- The working party starts in 2012
- Inventory testing & quality in ICT education 2014
- Testing & quality curriculum part of bachelor of science 2015
- Development of minor software engineering & testing 2016-2018
- First minor 2018-2019
- Second minor 2019-2020
- Pr & Marketing, Implementation is (still) running

History

Inventory current situation test education in the Netherlands

Respondents:

Type of university	Total number	Number of respondents
Universities	11	2
Universities of applied sciences	29	8

Current situation:

Subject	Year				Type of University
	1	2	3	4	
Introduction to testing	X				UAS
Testing techniques	X				UAS / U
Test organization				X	UAS
Test phases		X			UAS
Review of requirements		X			UAS
Test execution		X			UAS
Defect management		X			UAS
Model based testen				X	UAS / U
Testing & development methods				X	UAS
Testmethods				X	UAS
Testtypes					UAS
Development testplan					UAS
Testtools					U

UAS = University of Applied Science
U = University

Conclusions:

- No. of respondents around 20%
- Mixed picture
- General & common items are educated based on the response

The results

Definition of a test curriculum

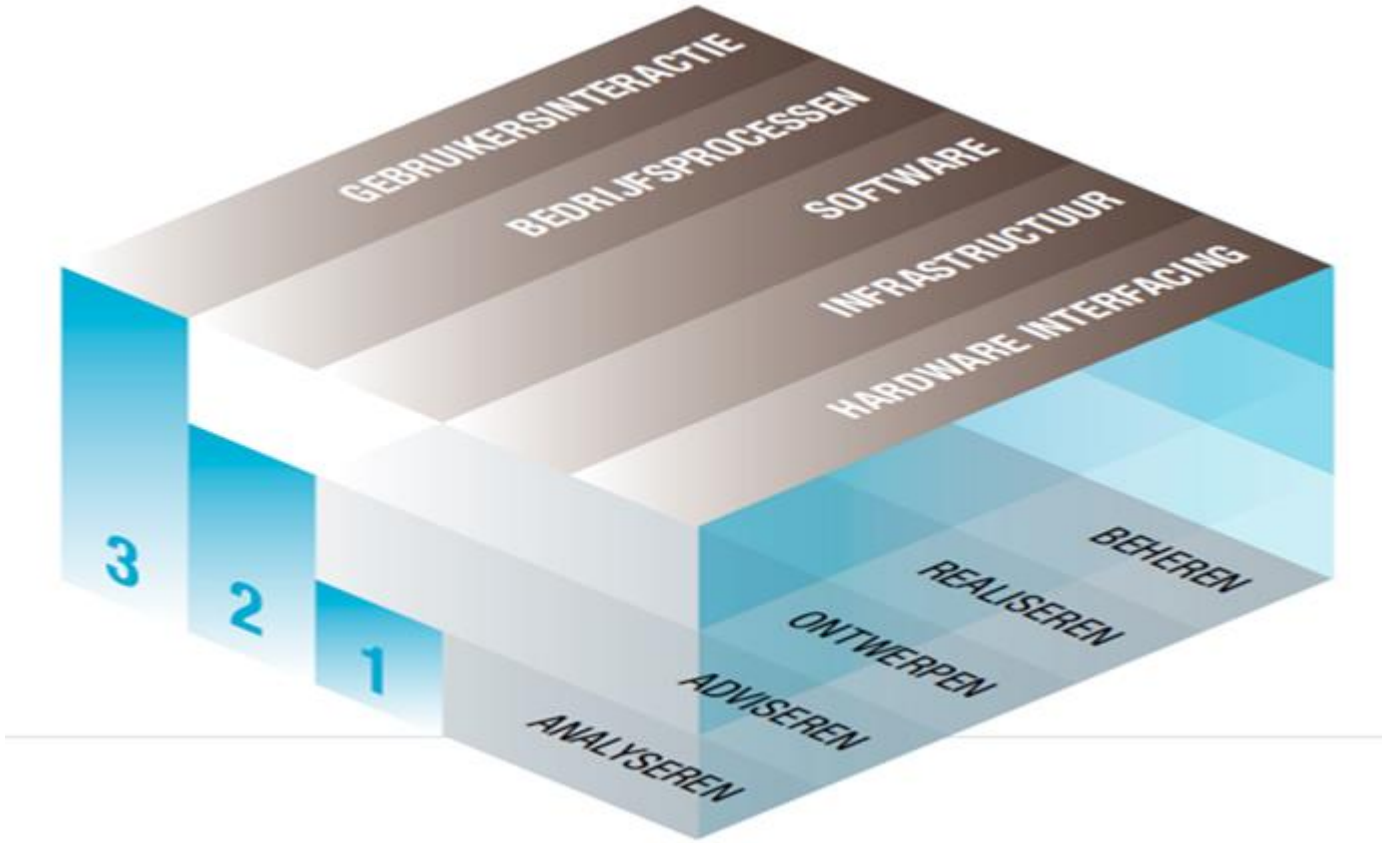
Curriculum based on generic description of the council of the Universities of Applied Science:

Structure of the curriculum is:

- Based on architecture layers of the E-competence framework
- Following the System Development Life Cycle (SDLC)
- Spreaded around 3 maturity levels
- Supported by literature list

Curriculum

- Embedded in the ICT curricula
- Not a separated program
- The idea is to teach quality when students are about to learn how to design for example a feature or to program a feature, like:
  - Reviewing the use case
  - White box testing a software component


GEBRUIKERSINTERACTIE
BEDRIJFSPROCESSEN
SOFTWARE
INFRASTRUCTUUR
HARDWARE INTERFACING
3
2
1
ANALYSEREN
ADVISEREN
ONTWERPEN
REALISEREN
BEHEREN

An example of the curriculum

	Analysis (G1)	Advise (G2)	Design (G3)	Realization (G4)	System management(G4)	General (G5)
User interaction	Review of the user interaction analysis Analysis of the stakeholders Determination of the completeness by hand of usability standard	Social skills	Review of the design Simulation of the interaction process Risk-based testing Definition of the required metrics Definition of the test approach Development of the required test cases Selection & implementation of the required tooling	Execution of the test cases Test reporting Tooling	Test ware management Configuration management Root-cause analysis Tooling	Evaluation test proces
	(B1)	(B2)	(B3)	(B4)	(B5)	(B6)
Business processes	Business process simulation Assessment of the compliance	Social skills	Process simulation by hand of the acceptance criteria Modeling/ Model checker Definition of the test approach Development of the required test cases Selection & implementation of the required tooling	Execution of the test cases Test reporting Acceptance procedure	Test ware management Configuration management Root-cause analysis Tooling Knowledge assurance Testing emergency procedure	Evaluation test proces
	(S1)	(S2)	(S3)	(S4)	(S5)	(S6)
Software	Determination of acceptance criteria Test roles Audit of requirements Risk-based testing	Social skills	Defining the test plan Development of the required test cases Selection & implementation of the required tooling Software metrics	Execution of the test cases Test reporting Acceptance procedure Application of Test tooling Metrics Test environment Assemble test data Test automation Determination of data quality	Test ware management Maintainability testing	Evaluation test proces
	(I1)	(I2)	(I3)	(I4)	(I5)	(I6)

Implementation

- Still going on
- Tremendous throughput time
- Curricula are revised not every year

Type	#US	#started	Type	#UAS	#started
U	11	0	UAS	29	11

Defining the minor / specialization on software testing

Goals of the minor:

- Develop a program to specialize students in software testing
- Separate part of their study
- Base for further certification
- All relevant elements of the field of software testing must be incorporated. Basic skills but also new developments
- Make use of the experience of experts in the field

Minor software engineer & testing, program

Minor Software Quality and Testing 2019/2020

	Part 1: Fundamentals				
	Part 2: Advanced				
Week		Monday - 23-09-2019	Tuesday - 24-09-2019	Wednesday 25-09-2019	Thursday - 26-09-2019
	Part 3: Specials				
Week		Monday - 21-10-2019	Tuesday - 22-10-2019	Wednesday - 23-10-2019	Thursday -24-10-2019
	Part 4: Internship				
Week		Monday	Tuesday	Wednesday	Thursday
13	18-11-2019				
14	25-11-2019				
15	2-12-2019				
16	9-12-2019				
17	16-12-2019				
18	6-1-2020				
	Resit / repair cases				
19	13-1-2020	resit exam 1	resit exam 2	resit exam 3	repair cases
20	20-1-2020	repair cases	repair cases	repair caes	repair cases

Minor software engineer & testing

2018-2019:

- 18 students
- 17 passed the exam
- Teachers, 80% originate from the working field
- Feedback of the students:
  - Evaluation of the minor: 8.0
  - Experts of the field highly appreciated
  - Broadening of the knowledge regarding software quality
  - Sometimes very theoretical
  - Experts of the field are very enthusiastic
  - As known 2 students started their career in software testing

Minor software engineer & testing

- 2019-2020:
  - 25 students
  - Teachers, 80% originate from the working field
  - The program is still running
  - Improvements are made regarding the schedule

Future work

- Enhancement of the test curriculum
- Securing of the test curriculum and the minor
- Further enhancement of the minor software engineering and testing
- Learning objectives must be tightened up
- Further implementation of the curriculum

Questions / Comments

identify
THE QUALITY DRIVER