

# Proposal for Software Quality Academic Program

---

Developed by: Shmuel Gershon

Main contributors: Michael Stahl, Ron Moussaffi (Intel); Dan Almog (Ben-Gurion University)

December 2012. Jerusalem, Israel

Slight updates and edits: August 2016, Michael Stahl

## **Brief**

This document is a proposal for an academic program focusing on Software Systems Quality and spanning undergraduate and graduate studies. The program is tailored to current and future needs, but not predetermined to a specific academic institute.

A large portion of software workers define their job within software quality and in an industry that relies on systems that get more complex and more crucial, a program as described below answers needs of individuals, companies and colleges alike. The goal of the program is to create professionals that have a better grasp of the quality concept, understand how the different actions in software creation affect the delivered quality and how to effectively manage a software process to minimize quality risks.

## Academic Scope

Graduates of the proposed program can follow a Major in Software Quality (*as part of a Business Management, Computer Sciences or Industrial Engineering degree*) by selecting the mandatory courses and minimal requirements for elective courses.

The classes can also be used as content for a Master in Software Quality degree, by the completion of necessary courses and the creation of a research thesis oriented by a relevant quality professional.

## Curriculum Overview

(See appendix for further details)

Classes proposed for a professional of software quality:

- **Mandatory Topics in Quality:**
  - Quality Concepts
  - Quality Management
  - Software Testing I
  - Software Testing II
  - Programming
  - Software Testing Lab
  - Project Management
  - Project Management Lab
- **Pre-Requisites for Specialization:**
  - Software Systems Design I
  - Software Systems Design II
  - Software System Design Lab
  - Statistics
  - Recommended Courses:
    - Operating Systems
    - Data Structures
- **Elective Topics in Quality:**
  - Security
  - Performance and DBs
  - Performance Test Lab
  - User Experience and Usability
  - Legal Aspects of Technology
  - SW Requirements
  - Test Automation Lab
  - Quality Contexts:
    - Financial Services, Regulated Environments, Medical Industry, Big-Data Systems, Web apps and SaaS

## Partnerships and Cooperations

The academic program will come to fruition through joint effort of the industry and academic entities. From the academy, colleges like JCE and HUJI in Israel, or even international players like the Florida Institute of Technology, can provide expertise about the creation and maintenance of such academic curricula. From the industry, field leaders like Intel, Microsoft, Google and IBM can bring expertise about practices and real-world needs, as well as much needed sponsorships and opportunities for research.

## Outlook and Impact

The proposed curriculum is beneficial for all the involved parties:

- Students will have a new area for specialization, increasing their future employment prospects and fitting individual interests
- Academic institutes that offer this curriculum will attract students who are interested in the subject and select it as a career path. Offering Masters Degrees opens the door for cooperation with the industry – including research grants.
- Sponsoring companies and the industry will benefit from better trained new hires for any software related positions as well as academic partners to conduct research with possible practical uses.

## Appendix A: Course Content Details

The courses mentioned in the “Curriculum Overview” section need to have their syllabus better defined and expanded. Below is a suggestion for the direction in which such courses should/could be conducted. The list of topics assigned for each course is by no means a final one – it’s for sure just a partial list.

### Essential Topics in Quality

Quality Concepts:	Dissertations and discussions on the idea of quality and its factors, extrinsic and intrinsic value, subjective and objective judgment, different philosophies of quality.
Quality Management:	Construct Validity, Standards, Metrics, Reviews, Knowledge acquisition products, Lifecycle Uncertainties.
Software Testing I:	Testing Concepts, Testing in the SW Lifecycle, the Test Process, Functional methods for testing, functional techniques, non-functional methods and techniques, test report, bug reports, Introduction to test tools and test automation.
Software Testing II:	Advanced test techniques, static test techniques, Unit Tests, Advanced Test Automation, Test Planning and Strategizing, Test Management.
Programing:	Basics of programing: control structures (selection & repetitions), data types, basic data structures, functions. The language does not matter that much although it should be a modern one (Some C or Java flavor seems reasonable).
Software Testing Lab:	Creation of test documents, creation and execution of tests, bug reporting and isolation exercises.
Project Management:	Project Management basics, Development Lifecycle, Software Development Methodologies, ALM tools.
Project Management Lab:	Creation or simulation of lifecycle management with selected ALMs, identification of perceived and real risks, mitigations and decision-making.

### **Pre-Requisites for Specialization**

Software Systems Design I:	Introduction to Systems and system thinking, types of systems, systems of systems, Models and maps, interaction models.
Software Systems Design II:	System modeling, Design Patterns, Design Practices, Designing within a team.
Software System Design Lab:	UML, Modeling Tools ( <i>e.g. Rhapsody</i> ), team exercises.
Statistics:	Computation and application of statistics, research techniques, probability, sampling distribution, estimation, statistical hypotheses, Design of Experiments (DOE).

### **Elective Advanced Topics in Quality**

Security:	Security lifecycle, threat models, penetration methods, vulnerabilities and exploits.
Performance and DBs:	Stress, Load and Performance, Design for Performance, Assessment of Performance, databases and optimization.
Performance Test Lab	Experience with load and test tools, relevant automation, test throughput, speed, and transaction rate, on DB and networking applications.
User Experience and Usability:	Human-Computer Interactions, Computer-Computer Interactions, Designing and creating user experiences, Usability Assessments, accessibility principles and laws, Usability test methods.

Legal Aspects of Technology:	Privacy, Intellectual Properties, License type and uses, Patents, Trademarks and Copyrights, Contracts and Agreements.
SW Requirements:	Introduction to software requirements, stakeholder identification, requirements gathering, software specifications, requirements under constraints, requirements review.
Test Automation Lab:	Industry automation tools. Creation, organization and maintenance of automation systems for software products.
Quality Contexts:	Different commercial environments and their effect in the management and assessment of quality. Contexts suggested: Financial Services, Regulated Environments, Medical Industry, Big-Data Systems, Web applications and SaaS.

## **Appendix B: Additional possible topics**

The following is just a list of additional topics that may be offered as mandatory or elective courses. This is a late addition to the document, so is not expanded further.

- User Interface testing
- Mobile applications testing
- IoT testing
- Computer vision application testing
- Agile development and test process
- Continuous Integration; DevOps
- Advanced Security testing