

## **Tárgyleírás**

**Tárgy neve: Software quality and testing**

**Tárgyfelelős neve: Kovács Attila**

**Tárgyfelelős tudományos fokozata: DSc**

**Tárgyfelelős MAB szerinti akkreditációs státusza: AT**

**Az oktatás célja angolul / Aim of the subject:**

- Promote efficient and effective communication by using a common vocabulary for software testing.
- Understand the fundamental concepts of software testing.
- Demonstrate an understanding of how different development and testing practices, and different constraints on testing, may apply in optimizing testing to different contexts.
- Contribute effectively to reviews.
- Use established techniques for designing tests at all test levels.
- Interpret and execute tests from given test specifications. Report on test results.
- Understand test management principles for resources, strategies, planning, project control, and risk management.
- Write and communicate clear and understandable defect reports.
- Understand the project factors that drive the test priorities and test approach.
- Understand the value that software testing brings to stakeholders.
- Appreciate how testing activities and work products align with project objectives, measures, and targets.
- Assist in the selection and implementation process of the testing tool.

**Knowledge on**

- The engineering principles that underpin the general knowledge of the IT field,
- The connections and concepts related to the above principles, especially in application development and quality control,
- The testing processes in the IT field,
- The special tools of the field.

**Abilities:**

- Able to apply general IT quality principles, facts, rules and contexts in the field of informatics.
- Able to apply formal and semi-formal test models of informatics.
- Able to perform testing tasks in the IT field.
- Able to use efficient techniques for designing tests.

**Attitude:**

- Understands and authentically represents the principles of the profession.
- Accepts the rules and ethical principles of the work and organizational culture of the IT profession.
- Seeks to collaborate with professionals in other fields.
- Takes into account the legal regulations of the IT field.

#### **Autonomy, responsibility:**

- They take responsibility for their professional decisions made in their IT-related activities.
- They undertake to meet deadlines and to have deadlines met.
- They bear responsibility for their own work as well as for the work of their colleagues they work together with in a project.
- Regarding mission critical IT systems, they can be entrusted with developing and operational responsibilities that are in accordance with their professional competencies.

#### **Major topics:**

- Basics of software testing, human error, defect, failure. Challenges of testing, test base, test condition, test case. Testing principles.
- The place and role of testing in the software development lifecycle. Sequential, evolutionary models. The V-model. Levels of testing.
- Iterative models, RAD, UP, agility. XP, SCRUM, Kanban.
- Basic testing lifecycle, testing planning and control. Risk analysis and estimation methods.
- From technical design to test execution. Entry and exit criteria.
- Human aspects of testing, test management. Independence of testing. Test team construction.
- Specification-based testing. Equivalence partitioning, boundary value analysis, decision tables. State transition testing, use case testing. Combinatorial testing.
- Structure-based testing. From statement coverage to path coverage. The cyclomatic measure.
- Static testing. Review techniques, static analysis.
- Supporting processes and tools. Configuration management, incident management, software maintenance.
- Test tools. Software quality. ISO 9000, SPICE, CMMI.
- Case study.

#### **Requirements and evaluation:**

- The exam conforms with ISTQB CTFL.

#### **Literature:**

- Graham, van Veenendaal, Evans, Black: Foundations of Software Testing (Thomson, 2007)

- Forgács, Kovács: Practical Test Design (BCS, 2019)
  - Galin: Software Quality Assurance: from theory to implementation (Addison-Wesley, 2004)
  - Rombach: A handbook of software and systems engineering (Pearson, 2003)
-