

## Tárgytematika / Course Description

### Rendszer és szoftvertesztelés

GKLM\_INTM057

**Tárgyfelelős neve /**

**Teacher's name:** dr. Csapó Ádám Balázs

**Félév / Semester:** 2020/21/2

**Beszámolási forma /**

**Assesment:** Vizsga

**Tárgy heti óraszám /**

**Teaching hours(week):** 0/0/0

**Tárgy féléves óraszám /**

**Teaching hours(sem.):** 12/0/0

---

### OKTATÁS CÉLJA / AIM OF THE COURSE

Összetett szoftverrendszerek fejlesztése során fontos feladat a folyamatos tesztelés, ami egyrészt lehetővé teszi, hogy a hibák felderítése és javítása időben és költséghatékony módon megtörténjen, másrészt pedig bizonyos szintű garanciát ad arra, hogy a szoftverrendszer kész terméké válásakor helyesen működik. A tárgy elvégzésével a hallgatók átfogó képet kapnak a rendszer- és szoftvertesztelés elméleti háttéréről és gyakorlatáról.

---

### TANTÁRGY TARTALMA / DESCRIPTION

Az előadások az alábbi témaköröket érintik:

- Bevezető: miért fontos a rendszer- és szoftvertesztelés. Miért nehéz a szoftvertesztelés. Szoftverek életciklusa, szoftverfejlesztés korszerű módszertanai (V-modell, prototípus modell, gyors alkalmazásfejlesztés, Agilis szoftverfejlesztés, Scrum, Extrém programozás).
- Rendszer- és szoftvertesztelés hierarchikus szintjei: unit tesztelés, integrációs tesztelés, rendszertesztelés és ellenőrző (acceptance) tesztelés. Alapfogalmak: tesztelés, verifikáció és validáció. Teszteset, teszt specifikáció, tesztkészlet, hibamodell, teszt folyamat, teszt lefedettség.
- Unit tesztelés fogalmai és módszertana. Tesztminőségi mérőszámok: utasítás lefedettség, ág lefedettség és út lefedettség. Tesztelés eszközei: statikus elemzés forráskód alapján, illetve forráskód és modell alapján. Bemeneti tér particionálása, határérték-analízis, regressziós tesztelés, ok-hatás analízis, use case alapú tesztelés, strukturális tesztgenerálás, hibabecslés, felderítő tesztelés.
- Gyakorlati példák: unit tesztelés tesztkeretrendszer segítségével. Példa keretrendszer Node.js és Python nyelvekhez.
- Stratégiák integrációs teszteléshez - tesztelés top-down és bottom-up szemmel. Rendszertesztelés módszerei: szolgáltatás tesztelés, mennyiségi tesztelés, stressz-tesztelés, használhatósági tesztelés, biztonsági tesztelés, teljesítmény tesztelés, konfiguráció tesztelés, megbízhatósági tesztelés, dokumentáció-tesztelés. Ellenőrző tesztelés.
- Tesztmenedzsment - a tesztelés szervezeti keretei. Incidens- és konfiguráció-menedzsment. Hibakövetés.

## **SZÁMONKÉRÉSI ÉS ÉRTÉKELÉSI RENDSZERE / ASSESSMENT'S METHOD**

A tárgy előadásokból áll, és írásbeli vizsgával teljesíthető.

---

### **KÖTELEZŐ IRODALOM / OBLIGATORY MATERIAL**

Ficsor L., Kovács L., Kuser G., Krizsán Z. "Szoftvertesztelés" ( <https://gyires.inf.unideb.hu/KMITT/c12/> ), 2004.

R. C. Martin. "Clean Code: A Handbook of Agile Software Craftmanship". Prentice Hall, 2009

B. Laboon. "A Friendly Introduction to Software Testing". 2016.