

### Obiective

- Utilizarea instrumentelor de management a testelor **TestLink** (vezi [Tutorial TestLink](#)).
- Proiectarea testelor folosind tehnici de testare **black-box**.
- Utilizarea platformei de testare **JUnit 5.x** (vezi [Tutorial JUnit5](#)).

### Cerințe

Să se realizeze următoarele task-uri:

| Task, puncte                         | Descriere task  |
|--------------------------------------|---|
| <b>[TestLink]</b><br><b>4 puncte</b> | <ol style="list-style-type: none"> <li>1. În timpul laboratorului 2, fiecare student își va crea pe platforma <b>TestLink</b> un cont cu userul de forma <b>xyir1234</b>, conform indicațiilor din <a href="#">Tutorial TestLink</a>.</li> <li>2. Fiecare student din echipă va fi asociat (de către cadrul didactic de la laborator) proiectului <b>PrjTTT</b>, corespunzător grupei TTT din care face parte studentul al cărui id SCS se va utiliza la gestionarea testelor (desemnarea studentului se face la alegerea echipei). De exemplu, pentru o echipă care are studenți din grupele 231, 232 și 237, dacă studentul ales face parte din grupa 231, atunci toți studenții echipei vor fi asociați proiectului <b>Prj231</b>. Se vor realiza următoarele task-uri în proiectul <b>PrjTTT</b>: <ol style="list-style-type: none"> <li>3.1. definiți <b>funcționalitatea F01</b>, corespunzătoare enunțului problemei (secțiunea <i>Requirement Specification</i>) prin crearea <b>xyir1234_ReqSpec</b> care conține requirement-ul <b>xyir1234_F01</b>;</li> <li>3.2. definiți planul de testare <b>xyir1234_BBT_TP</b> în cadrul proiectului <b>PrjTTT</b> (secțiunea <i>Test Plan</i>);</li> <li>3.3. creați suita de teste <b>xyir1234_BBT</b> și adăugați câte un caz de testare valid și unul non-valid pentru cerința creată (secțiunea <i>Test Specification</i>), folosind ECP și BVA. Astfel, pentru tehnicile de testare studiate se vor defini în Testlink doar 4 cazuri de testare, i.e., câte un caz de testare valid pentru ECP și BVA + câte un caz de testare non-valid pentru ECP și BVA);</li> <li>3.4. asociați cazurile de testare create la planul de testare <b>xyir1234_BBT_TP</b> și cerința <b>xyir1234_F01</b>;</li> <li>3.5. generați documentația aferentă din (secțiunea <i>Test Specification</i>, opțiunea <i>Generate Test Specification Document</i>) în format .docx.</li> </ol> </li> </ol> |
| <b>[Unit BBT]</b><br><b>5 puncte</b> | <p>Proiectați și implementați cazuri de testare cu date de intrare valide și non-valide folosind tehnicile de testare ECP și BVA pentru o metodă de pe nivelul <i>repository</i>, <i>service</i> sau <i>ui</i> care implementează <b>funcționalitatea F01</b>. Se vor utiliza:</p> <ul style="list-style-type: none"> <li>• pentru proiectarea testelor: fișierul <b>Lab02_BBT_TCs_Form.xls</b>; <ul style="list-style-type: none"> <li>○ minimum 3/4 cazuri de testare (valide și non-valide) folosind ECP;</li> <li>○ minimum 4 cazuri de testare (2 valide, 2 non-valide) folosind BVA;</li> </ul> </li> <li>• pentru implementarea testelor: platforma de testare <b>JUnit 5.x</b> (vezi <a href="#">Tutorial JUnit5</a>). Se vor utiliza cel puțin 5 adnotări distincte, frecvent utilizate, dar diferite de următoarele: @Before/AfterAll, @Before/AfterEach, @Test.</li> <li>• Pentru implementarea testelor: se va avea în vedere utilizarea șablonului <b>AAA (Arrange, Act, Assert)</b><sup>1</sup>.</li> </ul> <p>Se aleg <b>doi parametri ai metodei testate</b> și se definesc condiții asupra acestora. Condițiile (constrângerile) rezultă din specificațiile metodei. Pentru acești parametri se aplică ECP și BVA. La proiectarea testelor se consideră că parametrii de intrare neinvestigați aici au valori valide (<i>dummy objects</i>).</p>   |
| <b>[Git]</b><br><b>1 punct</b>       | <p>Se va actualiza conținutul repository-ului <b>Git</b> cu documentele elaborate în cadrul acestei teme:</p> <ul style="list-style-type: none"> <li>• în folderul <b>Docs/Lab02</b> fișierul <b>Lab02_BBT_TCs_Form.xls</b>;</li> <li>• în folderul <b>Docs/Lab02</b> fișierul cu documentația generată în TestLink;</li> <li>• pachetul cu teste implementate în Java folosind tehnicile ECP și BVA;</li> <li>• dacă este cazul, codul sursă modificat în urma depanării.</li> </ul>   |

### IMPORTANT!!!

**Înainte de predarea temei de laborator, fiecare echipă va completa formularul de la acest [link](#).**

<sup>1</sup> <https://blog.ncrunch.net/post/arrange-act-assert-aaa-testing.aspx>

<https://medium.com/swlh/an-experience-of-unit-testing-with-the-arrange-act-assert-aaa-pattern-part-i-53babd01c52b>

<https://medium.com/@nelson.wright/an-experience-of-unit-testing-with-the-arrange-act-assert-aaa-pattern-part-ii-5cee988654ab>

**Observații****[TestLink]**

- Realizarea acestui task se va începe în timpul laboratorului 2.
- Userul **xyir1234** asociat proiectului **PrjAAA** va avea rolul *leader*.

**[Unit BBT]**

- Pentru alegerea datelor de test se vor folosi următoarele tehnici:
  - a. identificarea claselor de echivalență (Equivalence Class Partitioning, ECP);
  - b. analiza valorilor limită (Boundary Value Analysis, BVA).
- În situația în care cazurile de testare alese evidențiază defecte/buguri (rezultatul așteptat nu este identic cu rezultatul obținut în urma rulării testelor), codul sursă se va depana și procesul de testare se va relua pentru TOATE cazurile de testare proiectate, i.e., re-testare + testare de regresie.

**Timp de lucru recomandat pentru rezolvarea temei de laborator ~ 4 ore/echipă.**

**Predarea temei de laborator****[TestLink]**

- Documentația generată de TestLink disponibilă în repository-ul git.

**[Unit BBT]**

Se vor elabora următoarele documente:

- [1]. fișierul **Lab02\_BBT\_TCs\_Form.xls** completat cu:
  - a. datele de identificare ale echipei (numele și prenumele, grupa);
  - b. titlul temei de laborator și data realizării;
  - c. enunțul temei de laborator;
  - d. pentru **funcționalitatea F01**, se vor completa tabelele din **F01\_ECP** și **F01\_BVA**. Având în vedere că nu toate cazurile de testare proiectate se pot implementa, în tabelul din **BBT\_TCs** se vor completa doar cazurile de testare pentru care există un corespondent în codul sursă de testare.
  - e. în **BBT-TCs** se va completa în tabelul **Statistics** numărul total de teste rulate, numărul de teste *passed*, numărul de teste *failed*, numărul de bug-uri identificate, dacă au fost eliminate sau nu și statistica testelor după re-testare și testare de regresie.
- [2]. codul sursă din repository-ul Git va conține:
  - a. codul sursă al aplicației testate și depanate;
  - b. implementarea testelor.

**Termene de predare**

| Săptămâna | Tema de laborator |                              |                     | Primul termen de predare | Ultimul termen de predare |
|-----------|-------------------|------------------------------|---------------------|--------------------------|---------------------------|
| S03       | L02.              | TestLink. Testare Black-box. | TestLink, JUnit 5.x | S05                      | S07* <sup>o</sup>         |
| S04       |                   |                              |                     | S06                      | S08* <sup>o</sup>         |

\*) Temele de laborator restante se pot preda în limita timpului disponibil.

<sup>o</sup>) Se pot preda cel mult două teme de laborator.