

# ДОМАЋИ ЗАДАТАК ИЗ ТЕСТИРАЊА СОФТВЕРА за школску 2016/2017. годину

Рок за предају решења домаћег задатка је: **23.01.2017. / 11.02.2017.**

**Корисни линкови и упутства за употребу алата за израду домаћих можете наћи на следећим линковима:**

- [IBM Functional Tester](#) - туторијал на српском
- [IBM Functional Tester](#) - званични линк

## **Задатак #1 - Тестирање веб апликације техникама црне кутије**

Потребно је функционално тестирати задате клијент-сервер апликације на оперативном систему [Windows 7 Professional x32](#) (виртуелна машина, са задатим алатом, дата уз ДЗ), коришћењем алата [IBM Rational Functional Tester 8.6.0.7](#) и [Selenium IDE](#).

Приложену вирт. машину са инсталираним пакетом *IBM Rational* можете преузети у Павиљону од администратора (величина око 26 Gb). Као апликацију за тестирање техникама црне кутије дате су студентске веб апликације, имплементиране у прог. језику PHP уз коришћење технологије AJAX и релационе базе података MySQL. Списак студената и задатака који су им додељени налази се на крају овог документа:

Пројектни задатак	Веб локација	Назив базе података
Пројекат 0	<a href="http://localhost/projekat0/pocetna">http://localhost/projekat0/pocetna</a>	projekat
Пројекат 1	<a href="http://localhost/projekat1/ProjektniZadatak">http://localhost/projekat1/ProjektniZadatak</a>	s_drustvo
Пројекат 2	<a href="http://localhost/projekat2/ProjektIP">http://localhost/projekat2/ProjektIP</a>	etfsport
Пројекат 3	<a href="http://localhost/projekat3/mojprojekat">http://localhost/projekat3/mojprojekat</a>	conference
Пројекат 4	<a href="http://localhost/projekat4/projectJune/home.php">http://localhost/projekat4/projectJune/home.php</a>	science_newsstand

Веб локацијама је могуће приступити из било ког веб прегледача (енг. *web browser*) након што се у виртуелној машини покрене апликација *Wamp Server*. Након покретања сервера, у доњем десном углу појавиће се зелена иконица **W**.

Базе података су већ унете. Преглед података у базама можете извршити такође из веб прегледача, коришћењем следећег линка:

<http://localhost/phpmyadmin>

Приступни параметри за MySQL, уколико сервер буде тражио су (*username: root* , и није постављен *password* - празан).

Пројектни задаци (спецификације задатака у PDF-у) које су студенти имплементирали потребно је детаљно прочитати пре почетка писања тест плана.

Све спецификације налазе се у пројектним директоријумима, на локацији:  
C:\wamp\www\projekatX  
где је X број задатка {0,1,2,3,4}

Применом техника класа еквиваленције, комбиновано са граничним случајевима, развити низ тест примера за задати део апликације.

У извештају који се прилаже као решење домаћег задатка потребно је навести:

- План тестирања са списком свих класа еквиваленције
- План тестирања са свим тест примерима (енг. *test case*) које сте осмислили. Дизајнирати све тест примере, имплементирати их и извршити, да би се добили одговарајући извештаји (логови).

Потребно је дефинисати приближно 60 класа еквиваленције и све класе покрити са најмање 25 тест примера, од чега мора бити 20\* са аутоматизованом имплементацијом (скрипт), који морају покривати нетривијалне тест примере, и најмање 5 тест примера реализовати са мануелном (ручном) имплементацијом.

Код аутоматизованих скриптова применити све врсте верификационих тачки. Приликом тестирања форми обавезно користити регуларне изразе у тест скриптовима. У неколико тест примера применити коришћење аргумената и параметризованих тестова објектом *Test Datapool*-а (на пример: читање података из неког CSV фајла).

\* Могуће је све тестове реализовати у алату IBM FT, или првих 10 аутоматизованих тестова имплементирати у алату IBM FT, а других 10 тестова у алату *Selenium IDE*.

## Решење домаћег задатка

Као решење домаћег задатка очекује се реализован тест план и извештаји (логови) добијени извршавањем тестова. Извештаје реализовати као HTML стране.

## **Задатак #2 - Тестирање техникама беле кутије**

Овај део домаћег задатка односи се на тестирање Јавиних програма методама беле кутије и GUI тестирање (тестирање графичког-корисничког интерфејса). Овај део се ради у окружењу *Eclipse IDE* и алату *Abbot* (самостално их инсталирати и покренути, може у оквиру исте ВМ из задатка 1). Програми за тестирање дати су на крају овог документа. Студенти, уз одобрење асистента, могу узети и неку сопствену Јава апликацију, која има графички-кориснички интерфејс и која мора да задовољи следеће услове:

- 1) да графички кориснички интерфејс има неколико Јава Свинг компоненти (командна дугмад, радио дугмад или чекбоксове, текстуална поља,...);
- 2) да улаз примера није фиксиран (нпр. да може да прима податке са стандардног улаза, фајла, из базе података или кроз параметре метода);
- 3) да апликација има неколико класа;
- 4) да у неким методама постоје условне наредбе и петље;
- 5) да постоји најмање једна петља са променљивим бројем итерација.

У том смислу задате примере можете сами изменити и прилагодити их потребним условима. Наравно, неопходно је да апликација исправно ради.

Сваки студент мора урадити САМОСТАЛНО домаћи задатак и мора изабрати адекватан пример да би му решење домаћег задатка било признато. Уколико пример не задовољи услове, то повлачи у старту негативне поене. Решења која имају потпуну или сличност у већој мери, дисквалификују студената на овом домаћем задатку, без могућности да се овај домаћи поново брани у овој школској години.

[10 роена] Тестирање белом кутијом

Потребно је развити минимални скуп тест примера који ће покрити исказе, одлуке, петље и елементарне услове у изабраном Јава програму. Користити алат *ECLEmma* 2.3.2 *CodeCoverage* (<http://eclemma.org/>) за извештавање о процентима покривености. Тестове је потребно реализовати употребом *JUnit* алата (<http://junit.org/>), верзија 4.9 или новија.

[6 роена] Mock тестирање

Коришћењем библиотека *Mockito framework* (<http://mockito.org/>) или *EasyMock framework* вер. 3.4 (<http://easymock.org/>) у јединичним тестовима које сте реализовали, у најмање три тест примера објекат који се тестира треба да буде замењен са другим mock објектом који симулира понашање реалног објекта.

Избор библиотека треба да буде према броју индекса студента:

- Mockito (студенти са непарним бројем индекса)
- EasyMock (студенти са парним бројем индекса)

#### [4 роена] GUI тестирање

Коришћењем алата *Abbot ver. 1.3.0* (<http://abbot.sourceforge.net/doc/overview.shtml>) написати четири показна тест примера који служе за тестирање неких GUI компоненти у апликацији (тестови треба да имају најмање по 2-3 *assert* наредбе).

#### Решење домаћег задатка

Као решење домаћег потребно је приложити:

- Извршни код изабране апликације у прог. језику *Java*
- Све јединичне тест примере дефинисане алатом *JUnit*
- Извештаје алата за покривеност кода *CodeCover* (у html/xml формату)
- Све тест примере у којима је коришћен *Mockito/EasyMock*
- XML скриптове алата *Abbot*

#### СПИСАК СТУДЕНАТА

Број индекса	Презиме и име	Задатак 1 (број пројекта)	Задатак 2 (назив апликације)
2009/0555	Бурсаћ Вук	0	Java01_Calculator.zip
2010/0537	Јовић Никола	1	Java02_SudokuSolver.zip
2010/0545	Јанковић Марко	2	Java03_Shuffle.zip
2012/0533	Богдановић Немања	3	Java04_Hangman.zip
2013/0177	Грујић Нина	4	Java05_TicTac.zip
2013/0520	Костић Стефан	0	Java01_Calculator.zip
2013/0536	Зечевић Лука	1	Java02_SudokuSolver.zip
2013/0619	Ралић Филип	2	Java03_Shuffle.zip
2014/0016	Станковћ Тамара	3	Java04_Hangman.zip
2014/0157	Милојевић Владимир	4	Java05_TicTac.zip
2014/0387	Равић Невена	0	Java01_Calculator.zip
2014/0463	Миловановић Страхиња	1	Java02_SudokuSolver.zip
2014/0658	Кузмановић Оливера	2	Java03_Shuffle.zip
2014/0663	Перчобић Ђорђе	3	Java04_Hangman.zip
2015/0689	Димовски Борис	4	Java05_TicTac.zip
2015/0720	Ефремов Благој	0	Java01_Calculator.zip
2012/0519	Станишић Димитрије	1	Java02_SudokuSolver.zip