11TH INTERNATIONAL CONFERENCE ON EDUCATION AND NEW LEARNING TECHNOLOGIES

EDULEARN 19

PEER REVIEW SOFTWARE EVALUATION

M. Petrov[1], D. Damyanov[1], A. Aleksieva-Petrova[2]

[1]Sofia University (BULGARIA)

[2]Technical University of Sofia (BULGARIA)

PALMA (SPAIN)
1ST - 3RD OF JULY, 2019

About Me

- Sofia University,
- Faculty of Mathematics and Informatics,
- Department of Software Engineering
  - Milen Petrov, milenp@fmi.uni-sofia.bg

- Director of Master Degree Program "Data protection in computer systems and networks"

Where is it?

Milen Petrov, Sofia University,
milenp@fmi.uni-sofia.bg

Oldest university in Bulgaria

Milen Petrov, Sofia University,
milenp@fmi.uni-sofia.bg

Faculty of Mathematics and Informatics

- Best place to learn Software Engineering and Computer Science in Bulgaria (according to the national rating system*)

Milen Petrov, Sofia University,
milenp@fmi.uni-sofia.bg

Prerequisite

- to attract students by R&D of new tools
- immersive learning and assessment tools
- double-blind peer review process in programming courses
- software is well accepted, but extensibility is difficult to achieve

Milen Petrov, Sofia University,
milenp@fmi.uni-sofia.bg

Goal

- As a step further in the development of the system for peer-review:
  - we propose a methodology and tools for **enhancing and evaluating the existing peer review software** system

Milen Petrov, Sofia University,
milenp@fmi.uni-sofia.bg

Motivation/Background

SDLC (Software development lifecycle)

- *"finding and **fixing a software problem** (defect) after delivery is often 100 **times more expensive than** finding and **fixing it during the requirements and design phase**".*

B. Boehm et. All (2005)

Milen Petrov, Sofia University,
milenp@fmi.uni-sofia.bg

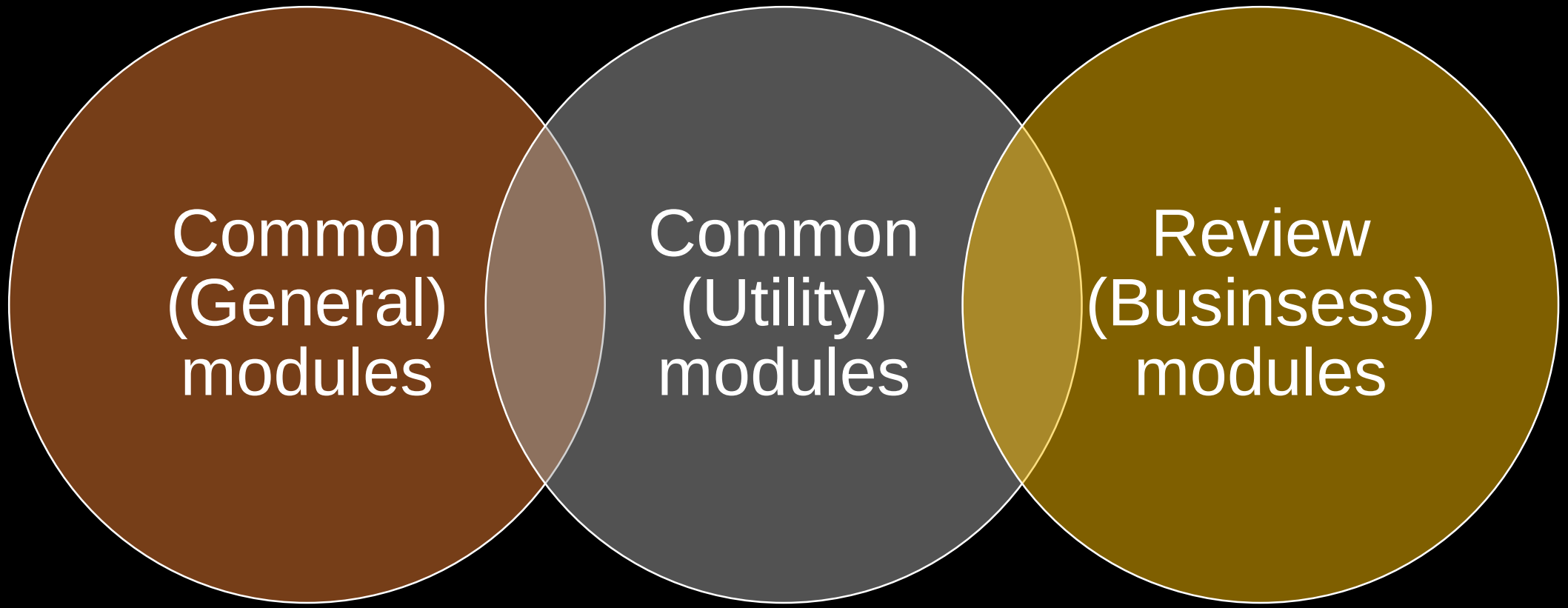
Motivation/Background (2)

- We can "***use social peer review (SPR),*** *which enables them (users) to* ***directly publish their work*** *within a web-based social network, where, instead of the traditional pre-publication peer review****, it can be evaluated and critiqued by the entire academic community***" [5]

(Matt, C., Hoerndlein, C., & Hess, 2017)

Milen Petrov, Sofia University,
milenp@fmi.uni-sofia.bg

Conceptual separation of concerns in target system



Milen Petrov, Sofia University,
milenp@fmi.uni-sofia.bg

Multi-tenant system and data

- The term "**software multitenancy**" refers to a software architecture where a single instance of software runs on a server and serves requests by multiple tenants.
- A tenant is a group of users who share a common access with specific privileges to the software instance.
- *With a multitenant architecture, a software application is designed to provide every tenant a dedicated share of the instance - including its data, configuration, user management, tenant individual functionality and non-functional properties. Multitenancy contrasts with multi-instance architectures, where separate software instances operate on behalf of different tenants* [6].

Milen Petrov, Sofia University,
milenp@fmi.uni-sofia.bg

Definitions

- Def 1: We use following definition for multi-tenant data as data to person or group of persons and/or group of groups with persons, which are processed and operated in model or sub-model of consistent data.
- Def 2: We define multi-tenant test data – as such data which can (eventually or partially) work with multi-tenant data and multi-tenant software. Multi-tenant test data should be consistent on both software API and user multi-tenant data. Multi-tenant data can be used in order to test the multi-tenant software.
- Def 3: We define multi-tenant test data definitions – data which can define and (eventually) work with multi-tenant test data.

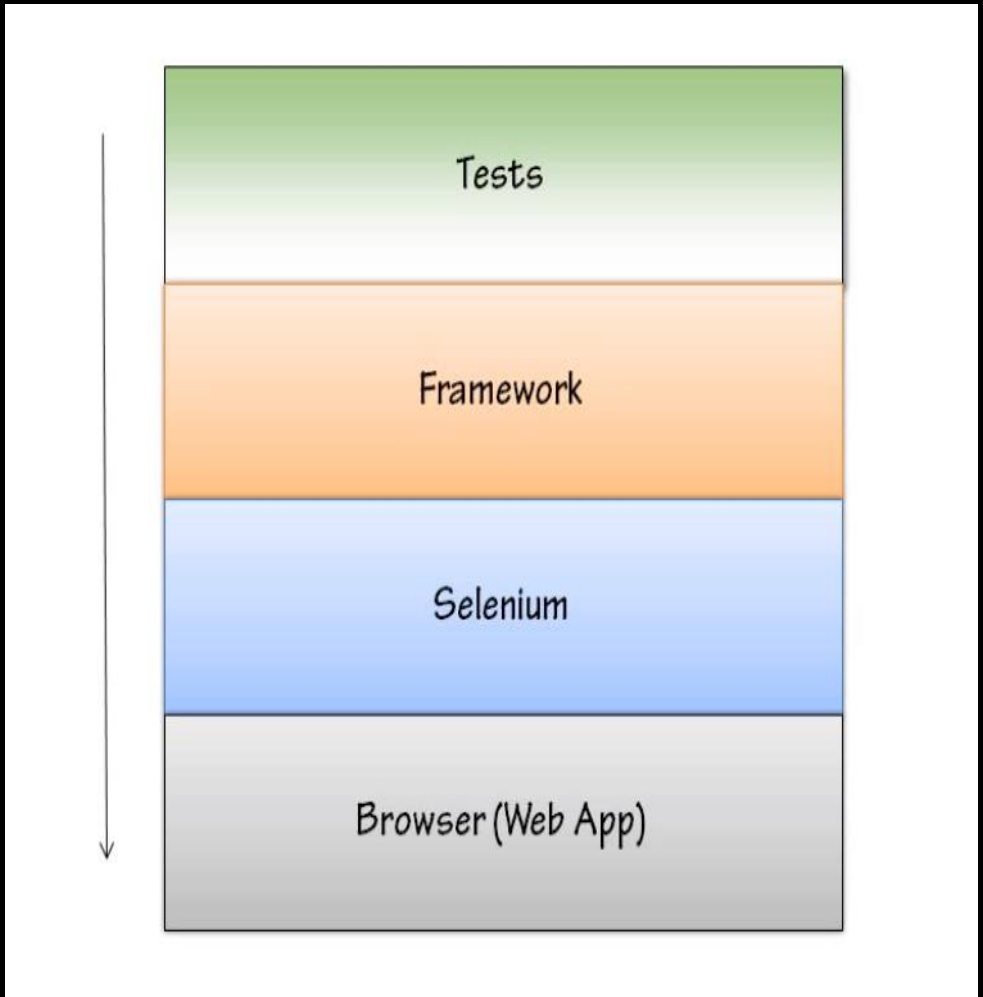
Milen Petrov, Sofia University,
milenp@fmi.uni-sofia.bg

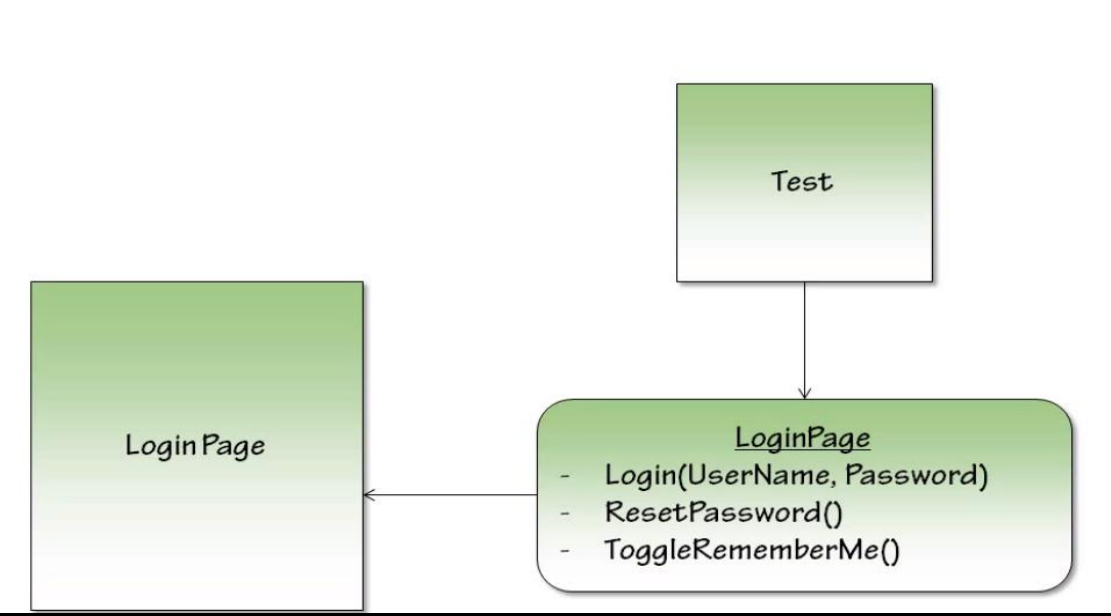
Evaluation and Application under test (AUT)

- **Application:** used for validation of our automation framework, is the Web application (Dynamic web site) for uploading and double-blind peer review of research papers written by the students
- **Users (students):**
  - 3rd grade Software Engineering BsC and
  - 4rd grade Computer Science (BsC)

Milen Petrov, Sofia University,
milenp@fmi.uni-sofia.bg

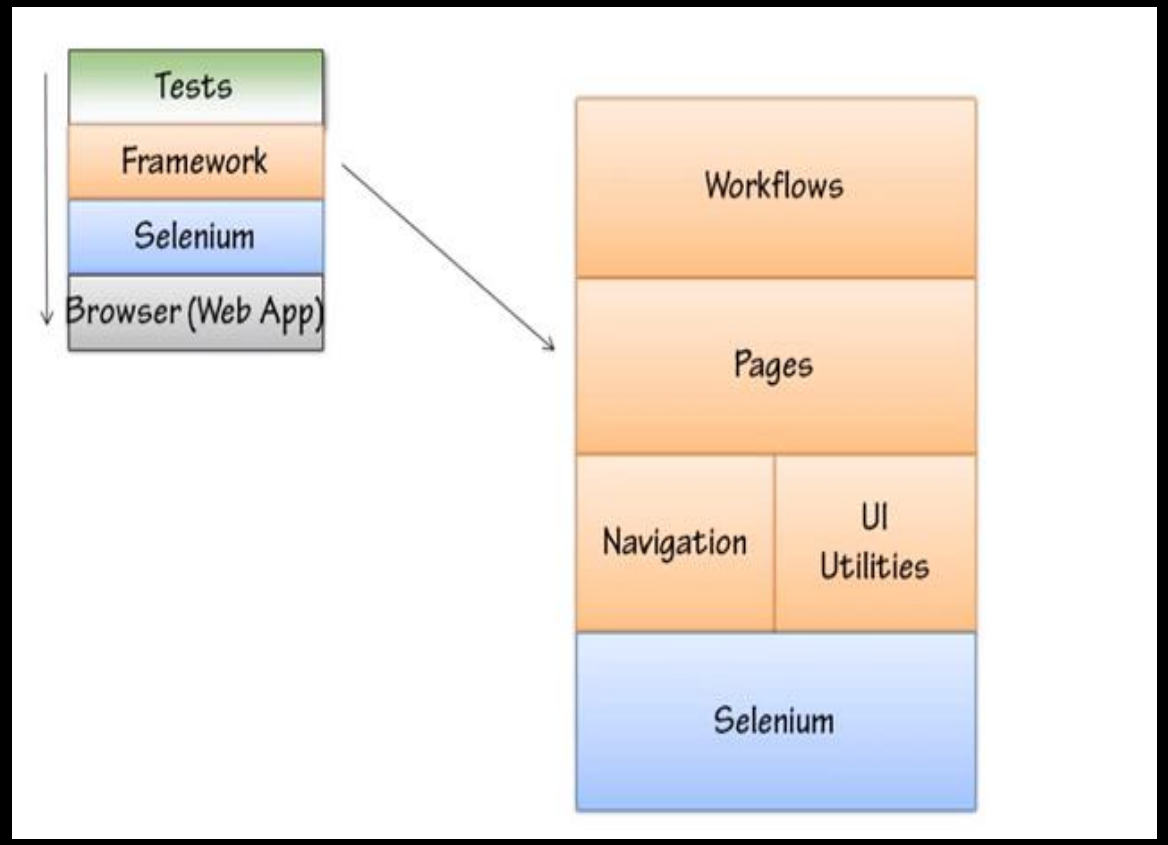
AUTOMATION TESTING FRAMEWORK

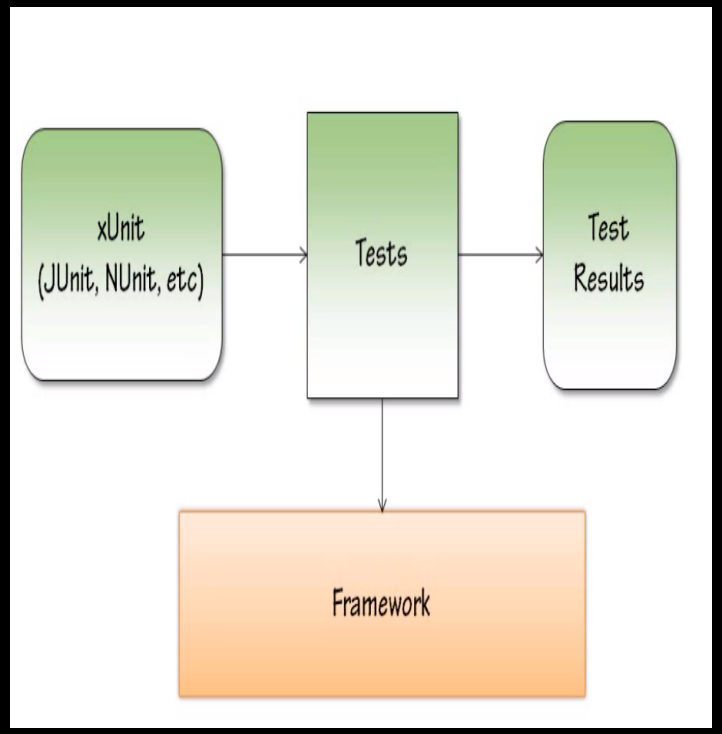




Milen Petrov, Sofia University,
milenp@fmi.uni-sofia.bg

Figure 4. (a) Layered architecture, used in the framework; (b) tests execution blocks.





Milen Petrov, Sofia University,
milenp@fmi.uni-sofia.bg

SOFTWARE SOLUTION FOR TESTS AUTOMATION

- *AutomationFramework* project is divided in four packages as follows *Pages*, *Navigation*, Workflows, and Selenium.
- **First package named "Pages"** is divided on few parts: {Dashboard, Login, PickThemes, Register, Review, UploadDocs}Page and ShellExecutor utility, depicted on fig. 5. We have two types of functionalities:
  - From one hand - specific *page modules*, represented by {*Dashboard, PickThemes, Review, UploadDocs*} and from other
  - General-purpose modules {*Login, Register, ShellExecute*}.
- **Second package named "Navigation"**, consists of *cross navigation modules*:
  - *Navigation* module (through specific page links)
  - *MenuSelection* module (through general menu)
- ***Workflows and selenium packages*** are dedicated to testing framework and tools.

Milen Petrov, Sofia University,
milenp@fmi.uni-sofia.bg

Logically tests for our target system are divided on three parts – process management, general modules and business modules.

I. Mgmt* process

- Bootstrap process*: prepare configurations, deploy, test, release
- Closing process: retrospectives, support, lessons learned (LL)

II. General modules

- Login, Register
- *Navigation, MenuSelector*
- ShellExecuter**

III. Business modules

- *Dashboard****
- PickThemes
- Review
- *UploadDocs****

Project 2: AutomationTests

Part 2.1. UI Tests

{UI}Tests={Register, Login, *Compare UI results*}

Part 2.2. Utility Tests

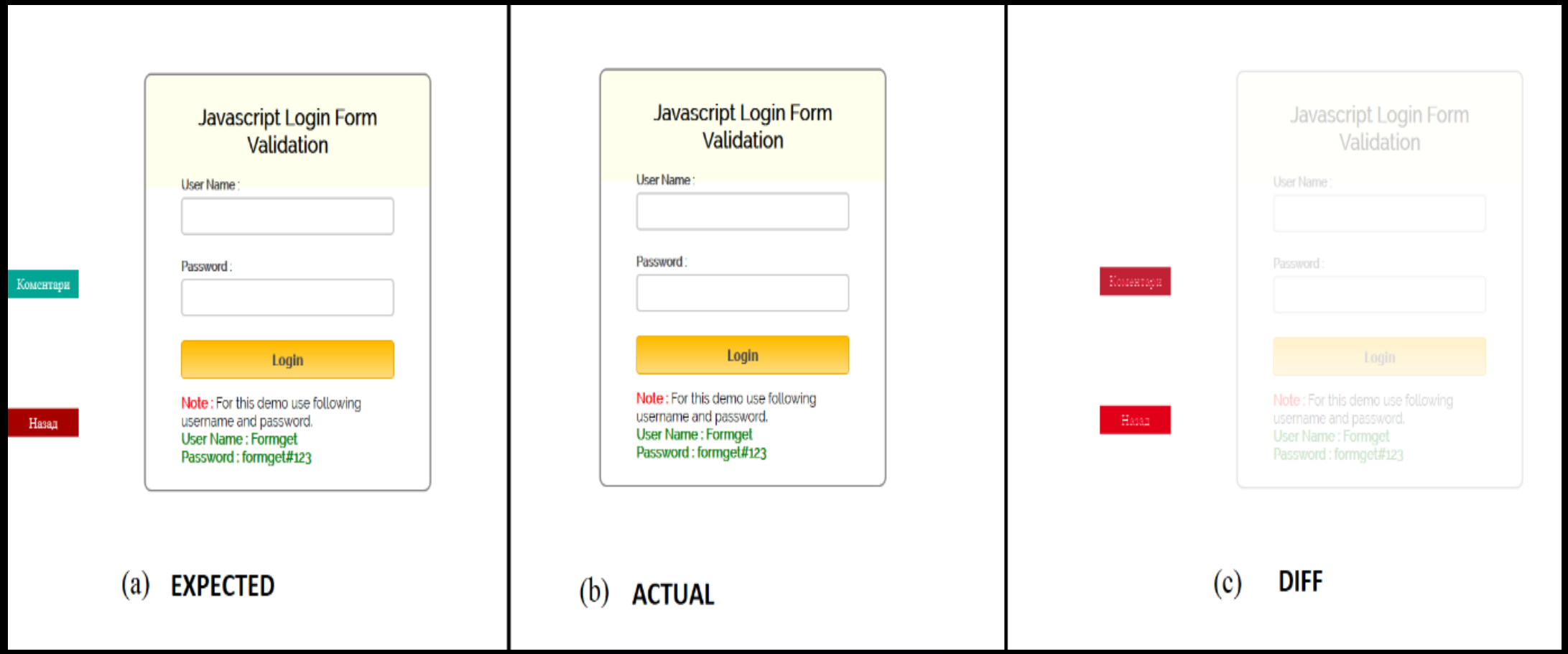
2.2.1. Process tests*

2.2.2. General utility tests**

2.2.3. General utilities with business use***

Milen Petrov, Sofia University,
milenp@fmi.uni-sofia.bg

Results (*Fig. 6. Visual interface comparison of User Interface Elements: (a) expected user interface by design; (b) result user interface after tests; (c) Difference between expected and actual UI.*

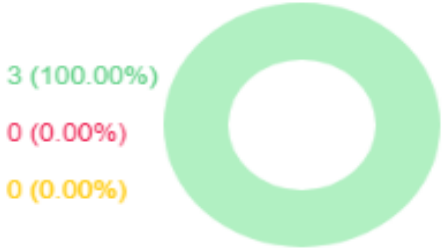


Milen Petrov, Sofia University,
milenp@fmi.uni-sofia.bg

Results from test execution of framework.

Results Summary

Pie View



Save graph

Tests Statuses	
Total	3
Executed	3
Passed	3
Failed	0
Inconclusive	0
Error	0
Warning	0
Timeout	0

Run Time Summary	
Start Time	2/16/2019 1:12:00 PM
End Time	2/16/2019 1:13:21 PM
Duration	1.34 minutes

Tests Details	
User	DESKTOP-QAB7TLC\Dido
Machine	DESKTOP-QAB7TLC
Description	

All Tests Group By Classes

Time	Status	Classes 1	Message	More
7/21/2014 10:56:45 PM		Tests.ReviewTests	3 Tests	Hide Tests

Time	Status	Test	Message	Owner	Duration
2/16/2019 1:12:00 PM	PASSED	Can_Set_Score (Data Row 0)			28.11 seconds
2/16/2019 1:12:00 PM	PASSED	Can_Set_Score (Data Row 2)			24.81 seconds
2/16/2019 1:12:00 PM	PASSED	Can_Set_Score (Data Row 1)			27.17 seconds

Milen Petrov, Sofia University,
milenp@fmi.uni-sofia.bg

CONCLUSIONS

- Developed testing framework can be used :
  - With the proposed AUT system, and also
  - with different application of converting database data(DB) to big data(BD), and
  - collect and enforce integrity of such systems as part of DB2BD project and
  - integrate it with different editors and data collectors.

Milen Petrov, Sofia University,
milenp@fmi.uni-sofia.bg

FUTURE WORK

- in general is planned to produce much more tests and evaluate results from these the tests.
- Develop and Integrate testing in cloud environment with security settings for collecting and handling tests as big data.
- Also try to apply the framework in several systems, which share the common concepts with tested system.

Milen Petrov, Sofia University,
milenp@fmi.uni-sofia.bg

ACKNOWLEDGEMENTS

- Research, presented in this paper was partially supported by the FNI-SU-80-10-138/15.04.2019, project of Sofia University “St. Kliment Ohridski” (Bulgaria) “Challenges of developing advanced software systems and tools for big data in cloud environment (DB2BD-2)” and by the contract KP-06-OPR/1 from 13.12.2018 project "An innovative software platform for big data learning and gaming analytics for an user-centric adaptation of technology enhanced learning (APTITUDE)", by Competition for financial support of basic research projects on societal challenges – 2018, funded by Bulgarian National Science Fund, Ministry of Education and Science, Bulgaria.

Thank you for your attention

Questions?

- Email: milenp@fmi.uni-sofia.bg
- Linkedin: /milen.petrov
- Twitter: @academika