NEW TRENDS IN SOFTWARE TESTING COURSE IN HIGHER EDUCATION

Olga Ristić, Marjan Milošević, Maja Radović, Sandra Milunović-Koprivica, Milan Vesković

Faculty of Technical Science in Čačak, University of Kragujevac, Serbia

Abstract

Software Testing is a fundamental course that is being studied at most IT faculties. However, in recent years, this subject is change because a large number of different types of software application have appeared (web, Android, iOS, ...). Nowadays, there are over a hundred types of software testing. Software Testing course becomes more important because production owner requests high-quality software. As this subject is studied at some IT faculties in Serbia, it is necessary to continually innovate the subject, but in some faculties is not in the curriculum. These faculties must take the initiative to make changes to the study program, in order for the students to be adequately educated and able to find a job in any stage of software development. This paper presents a review of faculties in Serbia that also have this course and faculties that do not have the Software Testing course in their curriculum. New trends need to be included in education so that IT students can be employed immediately after graduation.

Keywords: Software Testing course, education, students.

INTRODUCTION

The principle of software testing today differs greatly when it compared with those used a few decades ago. Employees who develop software application earlier performed tests on their own. Testing was based only on debugging the application. Later, in companies were employed software testers whose job was only based on testing. The test teams were formed within larger firms with more developers. Today, there are so many software firms that are exclusively dealing with software testing. Testing has become a significant process in the development of software products and is no longer the last stage in the software development life cycle (SDLC), but it is used in all developing stages. This means that the testers are included from the first phase of SDLC (user requirements), through the design and all other phases to the delivery and maintaining. It is therefore important to follow trends related to testing [1].

All that means, Software Testing (ST) must be even elective course in Information Technology (IT) higher education. In Serbian faculties who educate IT engineers in many curricula this course does not exist. That is the main disadvantage of our IT education.

HISTORY OF SOFTWARE TESTING

Software testing was starting to use when the first computer application was created. The word "testing" can be explained as an examination of whether something works properly, and whether the input values get the corresponding output values. Software testing was previously performed at the end of the code writing or in cases where the application written by the developer is complex the programmer most often implemented it after writing a particular module (function). However, in recent years with the rapid development of various types of applications (web, Android iOS, IoT ...) and different kind of devices running these applications, testing has become increasingly important. In the application development, the key role of the test bench is through all stages of SDLC. The software quality must be as good as it is in terms of the application functionality, as well as the satisfaction of the application users. This will be achieved only by proper testing.

In the last decade software testing becomes an important part of SDLC. Currently there are more than 100 different types of software testing [2]. What kind of testing will be carried out depends on many factors such as: the

complexity of the application, the type of application, whether it is necessary to be implemented on one type of device or on several different devices, etc. In simple applications, manual testing is dominant. With the increasing applications complexity, the automation of the testing process is becoming more intense, so the number of employees who will be employed on software testing increases every year.

NEW TRENDS IN SOFTWARE TESTING

Nowadays, testing has a lot of new flexible methodologies, deep integration with the development process, extensive use of test automation, an enormous set of technologies and tools, cross-functionality of the team (when the programmer and tester can do each other's work).

The quality assurance (QA) professionals have to rapidly adopt changes in the software testing industry to stay relevant. All this has contributed to the emergence of new trends presented in Figure 1 [3]. These trends must be including and learning in ST courses in higher education.
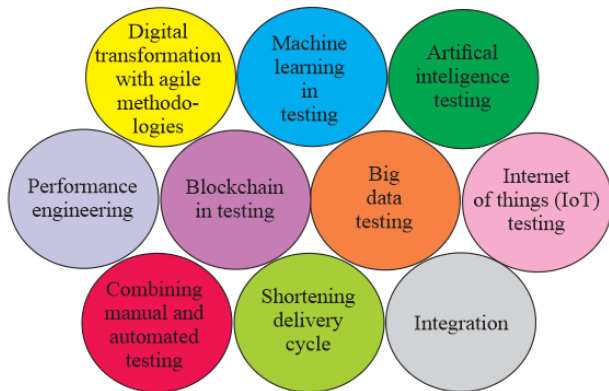

Fig. 1. *Software testing new trends*

With the advent of Agile and DevOps development technologies, the software development industry is undergoing major disruptions.

DIGITAL TRANSFORMATION WITH AGILE METHODOLOGIES

Business is undergoing digital transformation ever since data has become valuable in gaining insights. The latest addition to this trend is the adoption of agile methodology to undergo a digital transformation. Agile methodology helps to align digital transformation initiatives with business needs.

The agile team defines business challenges, objectives and use cases. In the agile approach, new features are delivered incrementally with each sprint. As digital transformation is an ongoing process, agile helps to deliver valuable outcomes (part of the software application) frequently for the business rather than waiting for a long time.

Main strategies in the digital world will be:
- Focus on customer experience, not on functional validation.
- Provide a different aspect of testing and end-to-end validation.
- Focus on innovation and continuous learning.
- Optimization and acceleration of testing.

MACHINE LEARNING IN TESTING

Machine learning is bringing revolutionary changes in software testing processes. Many mobile applications are difficult to test. Tester needs more time for testing on different devices, so the greater chance that testing will not properly be finished and bugs will be later found in the software. Consumers don't like these bugs, and they will not use it, which negatively affects their reputation. Machine learning teaches systems to learn and apply that software testing knowledge in the future. That's why machine learning makes software testers come up with more accurate results than traditional testing ever could. The probability of error is getting reduced and the time needed to perform a software is shortened. The amount of data that needs to be handled in the future application will increase. In software testing, machine learning can be used for:
- Test suite optimization – To identify redundant and unique test cases.
- Predictive analytics – To predict the key parameters of software testing processes on the basis of historical data.
- Log analytics – To identify the tests cases which need to be executed automatically.
- Defect analytics – To identify high-risk areas of the application for the prioritization of regression test cases.

ARTIFICAL INTELLIGENCE TESTING

The goal of using machine intelligence is to make testing easier and faster, just like with the traditional use of tooling to support any activity. Some tasks that couldn't be done before are now possible by using intelligent machines. Testing is used for gathering information about achieved benefits and remaining quality risks that is used to determine the confidence of stakeholders' system. This final judgment about confidence still has to be made by a human being, because the accountability for business processes can't be delegated to a machine, no matter how intelligent they may seem to be.

In the decision-making process, which includes standards, decisions can be based on facts rather than feelings and opinions. To believe that decisions made by machines powered by AI are correct, we need to test these systems. Such systems are already in use. Google, for example uses them to improve their products. The company is rethinking and has applied AI across all products to solve problems. For example, its Streetview automatically recognizes restaurants with the help of machine learning. Google is continuously testing and improving its machine learning using AI.

BIG DATA TESTING

Big Data is the high volume of data generated fast. In Big Data testing, testers have to challenge with automation, virtualization and large dataset. Tester works, usually, with an unstructured huge amount of data set. This type of testing focuses on performance testing and functional testing.

The quality of data is also a critical factor in big data testing. It is verified before the testing begins. The data quality is checked on the basis of various characteristics such as conformity, accuracy, consistency, validity, duplication, data completeness, etc.

INTERNET OF THINGS (IoT) TESTING

The rapid growth of products associated with IoT is having consequences in systems design. Many systems we now develop are parts of an IoT solution. Combining new properties together with all the connectivity options bring an infinite number of situations that IoT solution can be in.

There are more connected devices than ever before as IoT technology is gaining traction. IoT testing is conducted to test IoT technology based devices. The various types of testing for IoT systems are as follows:

- Usability Testing – To test the usability of an IoT system.
- Compatibility Testing – To check the compatibility of devices in the IoT system.
- Reliability and Scalability Testing – Simulation of sensors utilizing virtualization tools.
- Data Integrity Testing – To validate the integrity of data.
- Security testing – To validate the user authentication process and data privacy controls.
- Performance Testing – To test the performance of the connected devices in an IoT network.

PERFORMANCE ENGINEERING

Performance engineering soon will replace performance testing. The focus will be on analyzing how all the elements of the system work together. Executing performance test scripts will be changed. The elements of the system include hardware, software, performance, security, usability, configuration, business value and the customer. The main goal of performance engineering is collaborating and iterating all elements and delivering quickly high-quality software.

BLOCKCHAIN IN TESTING

Blockchain technologies make possible a digital asset unique and safely transfer ownership. With smart contracts, the conditions for the transaction are automated. Transaction history is kept in an audit trail and is immutable. What is stored in Blockchain is very safe: there is not one database that may be hacked to commit fraud. The individual blocks are all cryptographically secured, just like the identification of both the sending and the receiving party of every transaction.

There are many different Blockchain platforms available with different characteristics. The first step in a successful Blockchain implementation should, therefore, be to make the right choice:

Which platform with what characteristics will be the best enabler in reaching certain goals.

The first step in Blockchain implementation is to choose a Blockchain platform. Next step is to created software for interacting with the Blockchain solution such as mobile apps. Also, determine which services are proper for allowing connectivity from apps or external systems.

COMBINING MANUAL AND AUTOMATED TESTING

More and more quality assurance professionals are using a combination of manual and automated testing approaches to harness the benefits of both as well as to overcome their respective shortcomings. The manual testing is still dominating the testing industry. Although automated testing brings about efficiency in the testing process there are still certain areas such as usability and design which requires manual efforts.

The test automation helps the testing team to track and manage all testing needs and helps to delivered high-quality software. Specialized tools are used to control the execution of tests to compare the actual results against the expected results. Mainly the regression tests which require repetitive actions are automated. Automation tools are used for both functional and non-functional testing. Nowadays, there are more than 50 free automation software tools [4]. Sometimes it is difficult to choose the right one.

SHORTENING DELIVERY CYCLE

Rapid changes in technologies, platforms, and devices are putting pressure on software development teams to deliver the finished products faster and more frequently. Testing must be integrated with development to facilitate delivery. Software organizations are ready to invest more in improving their development and delivery processes by employing the right set of tools. The need for test management tools will increase to match up with shortening delivery cycles.

INTEGRATION

With shortening delivery cycles, there is a greater need for integration of various elements of a product development. To facilitate smart testing and analytics, the data is collected from disparate sources such as requirement management systems, change control systems, task management systems, and test environment. This implicates that there is a need for integrated tools which help in requirement management, task management, bug tracking and test management.

REVIEW OF IT UNDERGRADUATE PROGRAM IN SERBIA

The number of IT students at Serbian colleges is rapidly increasing. These occurrences barely have an impact on ST course, so in many faculties it remains aside. Most of the courses are based on software development, while mistakes and bugs in code are not being properly handled. No matter how experienced the programmer is mistakes can always occur. Software testing is crucial for software quality since costumer wants his product to be optimized and to work without errors.

List of all faculties in Serbia with corresponding undergraduate studies which provide ST course is presented in Table 1. In column No. of Students there is given a number of students who enrolled faculties in 2017 [5]. It can be concluded that only about 30% of all students who are studying undergraduate IT studies have ST course.

Therefore, most of the faculties cannot provide graduate students with knowledge in both software development and testing. Larger companies usually make separate sectors for software testing, which are better known as Quality Assurance (QA) sectors. The need for QA engineers (software testers) still continues to increase since there is a lack of qualified people for that position. Consequently, including ST course in as many as possible faculty programs will result in a greater number of QA engineers, which is beneficial for both companies and graduate students.

Increasing demands of end users and the complexity of applications created for them, the quality of software must be exceptional. Graduate students of IT studies must have abilities required for working at any stage of SDLC. Although ST course is a part of the curriculum at many faculties in Serbia for more than 20 years, there are still a lot of faculties where ST course is not being studied as a

separate subject. These faculties are displayed in Table 2. This can be a big disadvantage of the study program because proper knowledge of software testing cannot be achieved by learning as a part of other courses.

Approximately 3700 students attend undergraduate IT studies in Serbia [5]. About 50% of those students are graduating within the program completion deadline. Out of the total number of IT students, about a third of them have ST course included in their studies.

Authors of this paper propose that, due to the importance of this field, all IT study programs should have ST course at least as an elective subject. The above mention new trends in software testing must be included in the ST course program. Thus, graduate students would have the opportunity to apply for positions in the area of software testing. This would be beneficial for students and employers because students would be able to find a job in the field of software testing without additional training, and employers could hire qualified engineers without difficulties. □

Table 1. List of faculties which have ST course in IT undergraduate studies

No.	Faculty name and place	Undergraduate academic studies	No. of Students
1.	Faculty of Electrical Engineering in Belgrade	Software Engineering	180
2.	ICT in Belgrade	Internet Technology	40
3.	Faculty of Technical Science in Novi Sad	Engineering of Information Systems	80
		Software Engineering and IT	80
4.	Faculty of Science and Math in Novi Sad	Information Technology	80
5.	Technical Faculty "Mihajlo Pupin" in Zrenjanin	IT-Software Engineering	40
6.	Faculty of Science in Kragujevac	Informatics	100
7.	University of Novi Pazar	Software Engineering	40
8.	Department of Economics of the State University in Novi Pazar	Business Informatics	40
9.	Faculty of Information Technology in Belgrade, Metropolitan in Belgrade	Software Engineering	84
		Game Development	20
10.	RAF in Belgrade	Computer Science	25
11.	Faculty of Informatics and Computing, University Singidunum in Belgrade	Computer Science and Informatics	80
		Software and Information Engineering	50
12.	FINTECH, University Educons in Belgrade	Information Technology	60

CONCLUSION

To satisfy customer needs, the QA professionals' needs to stay updated on the latest testing trends. The only way to be immune to the disruptions in the software industry is to prepare for the future. The above-mentioned software trends will help testers in investing their time and efforts in acquiring the rights skills and testing tools.

Many changes have been made to the ST course in IT undergraduate studies. The challenge was to ensure that all these improvements and new trends met the objectives of the ST course. The main goal of this paper is to show the shortage of IT studies in Serbia without ST course. □

ACKNOWLEDGMENT

Table 2. *List of faculties which do not have ST course in IT undergraduate studies*

No.	Name and Place	Undergraduate academic studies	No. of Students
1.	Faculty of Electrical Engineering in Belgrade	Computer Engineering and Information Theory	250
2.	Faculty of Organization Science in Belgrade	Information Systems and Technology	330
		Information Systems and Technology-Distance Learning	100
3.	Faculty of Mathematics in Belgrade	Informatics	160
4.	Faculty of Electronic Engineering in Nis	Computers and Information Technology	250
5.	Faculty of Science and Mathematics in Nis	Computer Science	55
6.	Faculty of Technical Science in Novi Sad	Computing and Automatic	240
		Information Engineering	60
		Applied Software Engineering	160
		Animation in Engineering	60
		Information Technology	35
7.	Technical Faculty "Mihajlo Pupin" in Zrenjanin	Information Technology	80
		Information Technology Management	25
8.	Faculty of Science in Kragujevac	Computer and Software Engineering	60
9.	University of Novi Pazar	Computer Engineering	109
10.	Faculty of Science in Kosovska Mitrovica	Informatics	40
11.	Faculty of Technical Science in Kosovska Mitrovica	Electrical and Computer Engineering	80
12.	**Faculty of Technical Science in Cacak**	**Information Technology (from 2014 accreditation)**	**80**
		Electrical and Computer Engineering	80
13.	Faculty of Information Technology in Belgrade, Metropolitan	Information Technology	84
		Information Systems	66
		Computer Engineering	25
		Computer Design	20
14.	Faculty of Informatics and Computing, University Singidunum in Belgrade	Information Technology	50
15.	Faculty of Economics and Engineering Management in Novi Sad	Informatics	50
16.	Faculty of Information technology – ALFA, University in Belgrade	Information Technology	25
17.	Faculty of Mathematics and Computer Science –ALFA, University in Belgrade	Informatics	25
18.	Faculty of Computer Science, Megatrend	Informatics	75

REFERENCE

[1] Tom van de Ven, Rik Marselis, Humayun Shaukat: Testing in digital age, 2018, ISBN 978-90-75414-87-5
[2] https://www.guru99.com/types-of-software-testing.html [Access 5 October 2018]
[3] https://www.ijirset.com/upload/2016/april/99_Software.pdf [Access 6 October 2018]
[4] https://www.guru99.com/testing-tools.html [Access 5 October 2018]

[5] Šošić Stefan, Ristić Olga, Mitrović Katarina, Milošević Danijela: Software Testing Course in IT Undergraduate Education in Serbia, Proceedings of 7th International Scientific Conference Technics and Informatics in Education - TIE2018, Čačak, Serbia, 25-27th May 2018, ISBN: 978-86-7776-226-1, pp. 224-229.