8th International Scientific Conference
Technics and Informatics in Education
Faculty of Technical Sciences, Čačak, Serbia, 18-20th September 2020

Session 2: *IT Education and Practice*

Professional paper
UDC: 004.42:005

Adaptivity of Web Applications – Case of Preschool Web Portal for Interinstitutional Data Integration and Analysis

Ljubica Kazi [1*], Dijana Karuović[1], Dragica Radosav[1], Tatjana Lojović[2], Aleksandra Kalezić-Vignjević[3], Olga Lakićević[3]
[1] University of Novi Sad, Technical Faculty "Mihajlo Pupin" Zrenjanin, Serbia
[2] Preschool Institution, Zrenjanin, Serbia
[3] Center for Interactive Pedagogy CIP, Belgrade, Serbia
[*] ljubica.kazi@gmail.com

Abstract: *Adaptivity and adaptability of software has become one of most important issues, since changes are the essential aspect of modern agile approach to software development. It is very important to address types of changes and how they could be anticipated and supported in software as a product. Therefore, the adaptability and adaptivity of software rise as an important feature that needs to be planned, integrated and maintained during the software lifecycle. Aim of this paper is to present a case study which demonstrates the adaptivity and adaptability of a web portal, which has been developed for the particular needs of preschool management. This portal enables data collection and integration from various institutions, data processing and visualization of results, in aim to support strategic management decision making. This paper specially emphasizes adaptivity (automated self-adaptation of software) and adaptability particular aspects that were implemented in the solution - graphical, functional and platform-related.*

Keywords: *adaptivity, adaptability, web application, preschool, data integration, data analysis*.

1. INTRODUCTION

Changes are inevitable in modern software development. They are constant due to [1]: refinements in user requirements, infrastructure/ technology advancements, organizational and legislation transformations, which encourage appropriate software improvements. Frequent response to clients' needs and changes is key aspect in modern agile [2] software development.

Having human role and productivity in software development and maintenance, it is of a great importance to consider software to be adaptable and adaptive. The crucial distinction between these two terms is in the level of software self-adaptation [3], where being completely "adaptive" includes ability to perform automated self-adaptation (including behavior and structure), as a response to changes in working environment.

This paper presents results in creating an adaptive software solution – a web portal for preschools that enables collection of data from various sources, i.e. institutions, as well as the data processing and visualization. The purpose of the web portal is to be the support to strategic management, particularly in resources planning. Aim of this paper is to focus on specific aspects of the solution's adaptivity – challenges and implementation of this important feature.

The rest of the paper is organized as follows: section two provides theoretical background, section three presents related work, section four describes the developed web portal solution and section five specifies details regarding implemented adaptivity of the software. Finally, section six provides conclusions.

2. THEORETICAL BACKGROUND

According to [4][5], adaptive software is defined as a software that dynamically adjusts its' behavior during run-time, as an answer to changes in working environment. In [3], self-adaptive software is defined as a system with closed feedback, where the control information is gained from the system itself (the "self" component - states of the system, i.e. elements of the software architecture) and the working environment (the „context"– everything that is in the operative environment and that affects the properties and behavior of the system).

Figure 1. presents key elements and the process of self-adaptation, which includes sensors to detect changes in environment, monitoring process over

the events in the working environment, detecting the change, requesting and deciding to make change and acting upon decision by initiating effectors to perform the change.

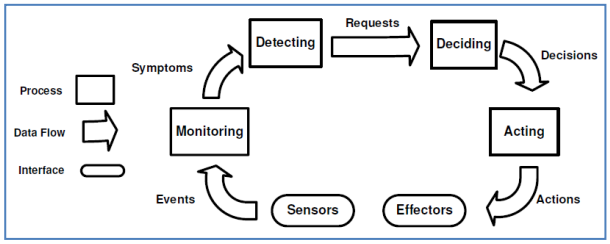


Figure 1. *Key elements and process of self-adaptation [3]*

Figure 2. presents key functional aspects of self-adaptive software, such as:

- *Primitive level (self-awareness and context awareness),*
- *Major level (self configuring, self-optimizing, self-healing, self-protecting),*
- *General level (self-adaptiveness).*

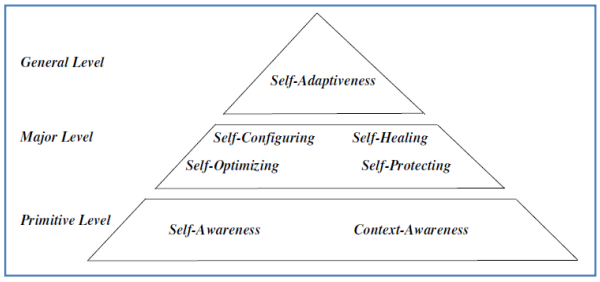


Figure 2. *Hierarchy of key features of self-adaptive software [3]*

Figure 3. presents two types of software regarding the level of automation in the adaptivity:

- Adaptive software (i.e. self-adaptive software) – internal approach where the adaptation mechanism is within the software solution itself,
- Adaptable software – adaptation mechanism is a module external to the core functional software itself.

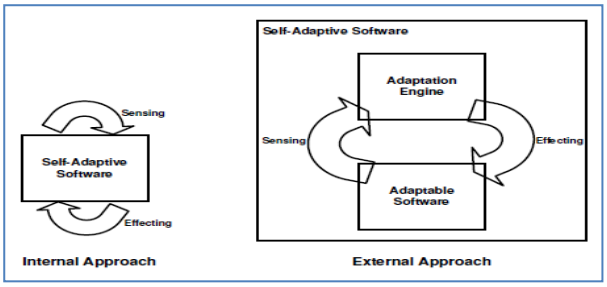


Figure 3. *Adaptive and adaptable software [3]*

3. RELATED WORK

Software changes generally could be positioned in:

1. Standards related to software maintenance (ISO/IEC 14764), with four categories: corrective, adaptive, perfective and preventive.
2. Software evolution, where Lehman [6] has defined eight laws that emphasize continual adaptivity of software, increasing complexity and broadening functionality, according to user needs and requirements.

Software changes management, within the software development lifecycle, was one of emerging areas in scientific research [1] and practice. Having software life-cycle in focus, starting with requirements changes anticipation [7], software changes are closely related to software evolution. Results in software evolution monitoring could be the basis to predict future requirements changes [8]. Case studies were analyzed to determine causes and effects of software requirements changes [9]. One of the main aims to encourage adaptivity of software is to enable software system longevity [10]. i.e. to have long living software system.

In the context of adaptive web-based information systems [11], one of the adaptivity aspects is related to the software user. The system user models enable adjustments to user profiles (created according to the user's knowledge, behavior and history). In that context, there are usually two types of software adaptation:

1. Adaptability– adjustments of the statical user profile. User is categorized to a group/profile, at the beginning of the software usage.

2. Adaptivity– multiple adjustments to "dynamic user profile", during software system work and usage. User characteristics could change, which requires constant monitoring of user behavior. Dynamic adaptation of the software is performed according to changes of the user behavior.

Research and development in software adaptability and adaptivity contributed with theoretical concepts, supported by practical results. The terms proposed in research work in this area are introduced as: pure adaptivity vs. intelligent adaptation [12], i.e. micro-adaptation (self-adaptive) vs. macro-adaptation (human-action dependent adaptation) [10]. Adaptation is analyzed from the user perspective (adaptive interfaces), operating environment and working context, as well as from the perspective of software architecture [10] [12].

Software adaptation is analyzed within different programming environments and approaches, such as interactive programming [12], rule-based systems [10] [13], model-based systems [14] [15], graph-based adaptivity [10], while implemented with software agents [13] [14].

Special concerns are focused on run-time adaptability [10] [15] and software evolution [15].

Adaptability and adaptivity of software have found particular applications in certain software working environments, where non-functional uncertainty (e.g. response time) could affect the software functionality [16], especially with possible dependency on fault-prone software components [17]. Particular applicative domains emphasize adaptivity as a critical feature, such as aviation (airplane software) [18] and legislation (law-related software) [19].

4. THE PRESCHOOL WEB PORTAL SOLUTION

Web portal for preschool strategic management was developed by joined team from Technical Faculty "Mihajlo Pupin" Zrenjanin, Serbia, Preschool institution Zrenjanin, Serbia and CIP center Belgrade, Serbia in year 2018. Initial focus was to support resources planning in preschool children population enlargement. Particularly addressed problem was determining and supporting children not included in preschools work. The goal of the project was to determine number of those unsupported children, by integration of relevant data from various institutions.

The business process model, for determining non-included children, was presented in [20]. That was the basis for software design of web portal [21]. Design was described with architectural, functional and data-related aspects. Implementation of this web portal included software quality- related ISO standards. Mapping of ISO standards-related software quality elements to elements of software solution was described in [20].

Developed solution represent a centralized web portal where institutions, related to the problem of non-included children, periodically enter relevant data. Web portal enables integration of these data and processing in aim to compute derived data, predictions and graph-based visualizations.

Web application was developed by using native PHP/MySQL without using frameworks and it is available: http://www.predskolskazr.edu.rs/projekat/index.php. The same application is also presented with a demo version, with illustrative amount of data http://www.predskolskazr.edu.rs/demo/index.php . The demo version has all the functionality as a regular version, but only has previously prepared illustrative amount of data, to be presented to potential users at presentations and usage trainings.

The first page of the web application is presented at Figure 4 and other relevant user-interface screens are given from the demo version at Figure 5,6,7 and 8.

Functional aspect of the developed web portal include software functions for three types of users:

- *Administrative user role (Figure 5)* - maintains data in coding tables (regions, institutions, data categories), registers institutions and employees from these institutions as software users, assigns institutions to particular data categories according to their responsibilities, defines computing formulas (combine data from coding tables, raw data entered by institutions employees and other computed/derived data).

Figure 4. First page of preschool strategic management web portal

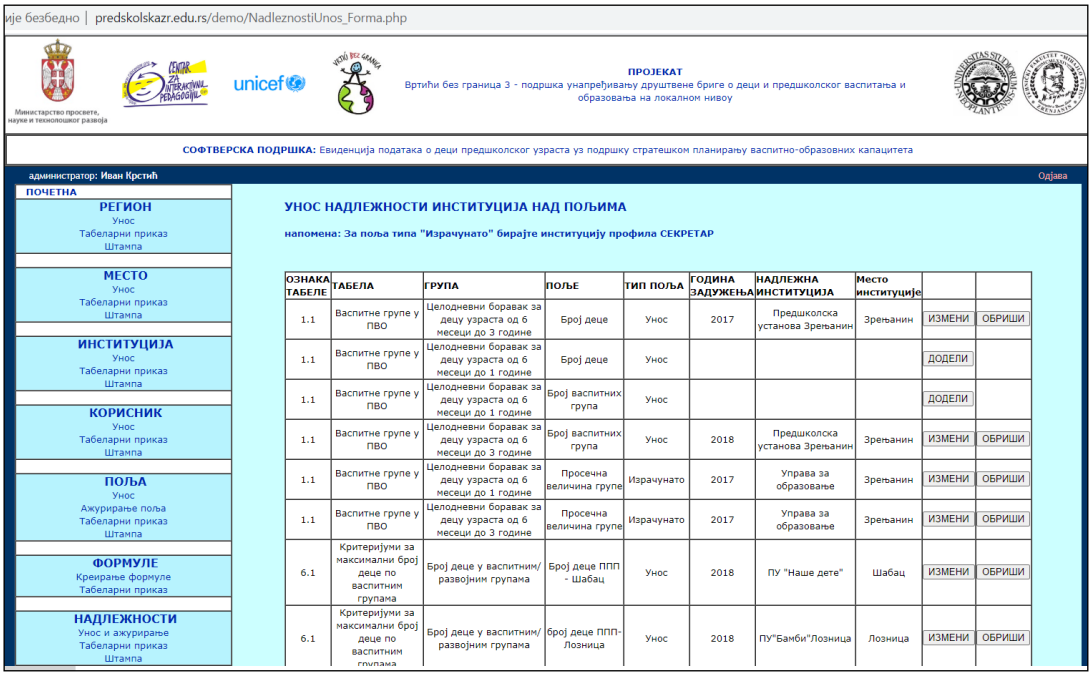


Figure 5. Assignment of data categories to the responsible institutions within administrative role in preschool strategic management web portal

- *Institutions" employee user role (Figure 6)* - engaged to enter raw data that are assigned to particular institution, as their responsibility.
- *Secretary (data analysts and manager) user role (Figure 7)* - initializes data collection by defining data entry period, monitors data entry progress, review the data collected and control missing data, archives data, performs data processing, prediction and visualization with graphs. Data processing is performed by combining raw data from institutions with local and national norms (i.e. referent values), by using previously defined formulas. Processed data are used for further analysis with data prediction (simplified assuming to have a linear function as a background, in this version of software). Visualization shows trends based on previously computed data prediction and it is presented in the form of line/bar type of graphs.

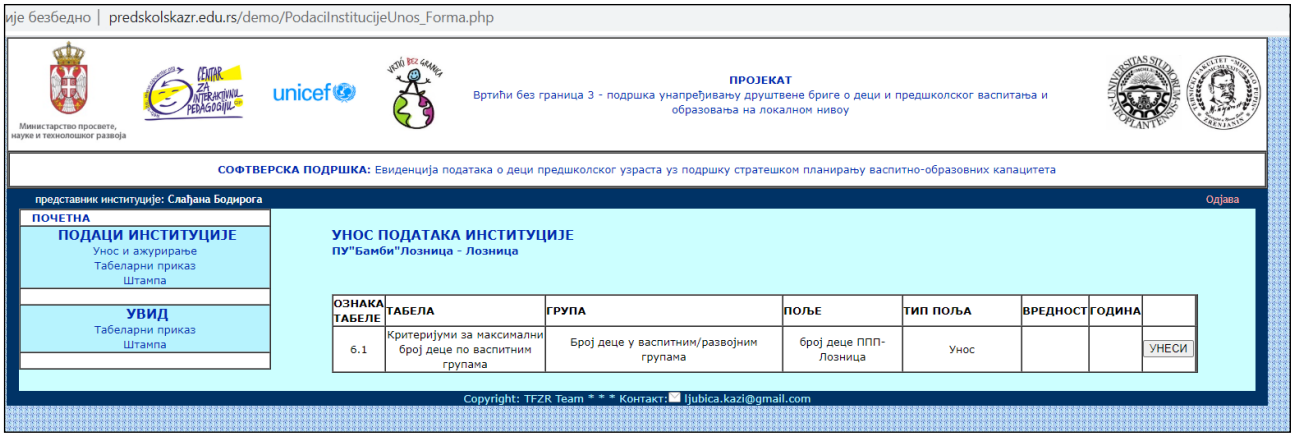
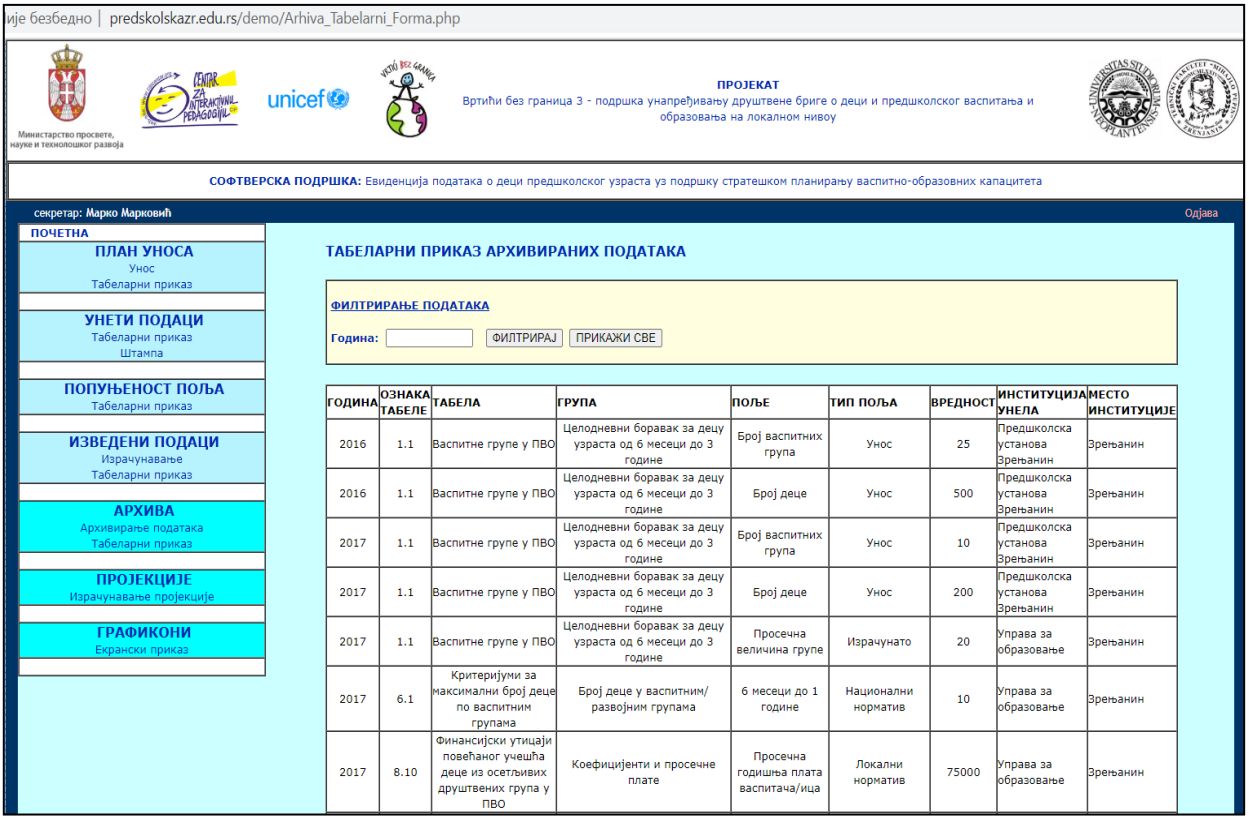


Figure 6. Page for assigned data entry within institution employee role of preschool strategic management web portal



ГОДИНА	ОЗНАКА ТАБЕЛЕ	ТАБЕЛА	ГРУПА	ПОЉЕ	ТИП ПОЉА	ВРЕДНОСТ	ИНСТИТУЦИЈА УНЕЛА	МЕСТО ИНСТИТУЦИЈЕ
2016	1.1	Васпитне групе у ПВО	Целодневни боравак за децу узраста од 6 месеци до 3 године	Број васпитних група	Унос	25	Предшколска установа Врењанин	Врењанин
2016	1.1	Васпитне групе у ПВО	Целодневни боравак за децу узраста од 6 месеци до 3 године	Број деце	Унос	500	Предшколска установа Врењанин	Врењанин
2017	1.1	Васпитне групе у ПВО	Целодневни боравак за децу узраста од 6 месеци до 3 године	Број васпитних група	Унос	10	Предшколска установа Врењанин	Врењанин
2017	1.1	Васпитне групе у ПВО	Целодневни боравак за децу узраста од 6 месеци до 3 године	Број деце	Унос	200	Предшколска установа Врењанин	Врењанин
2017	1.1	Васпитне групе у ПВО	Целодневни боравак за децу узраста од 6 месеци до 3 године	Просечна величина групе	Израчунато	20	Управа за образовање	Врењанин
2017	6.1	Критеријуми за максимални број деце по васпитним групама	Број деце у васпитним/ развојним групама	6 месеци до 1 године	Национални норматив	10	Управа за образовање	Врењанин
2017	8.10	Финансијски утицаји повећаног учешћа деце из осетљивих друштвених група у ПВО	Коефицијенти и просечне плате	Просечна годишња плата васпитача/ица	Локални норматив	75000	Управа за образовање	Врењанин

Figure 7. Data archive page of secretary role in preschool strategic management web portal

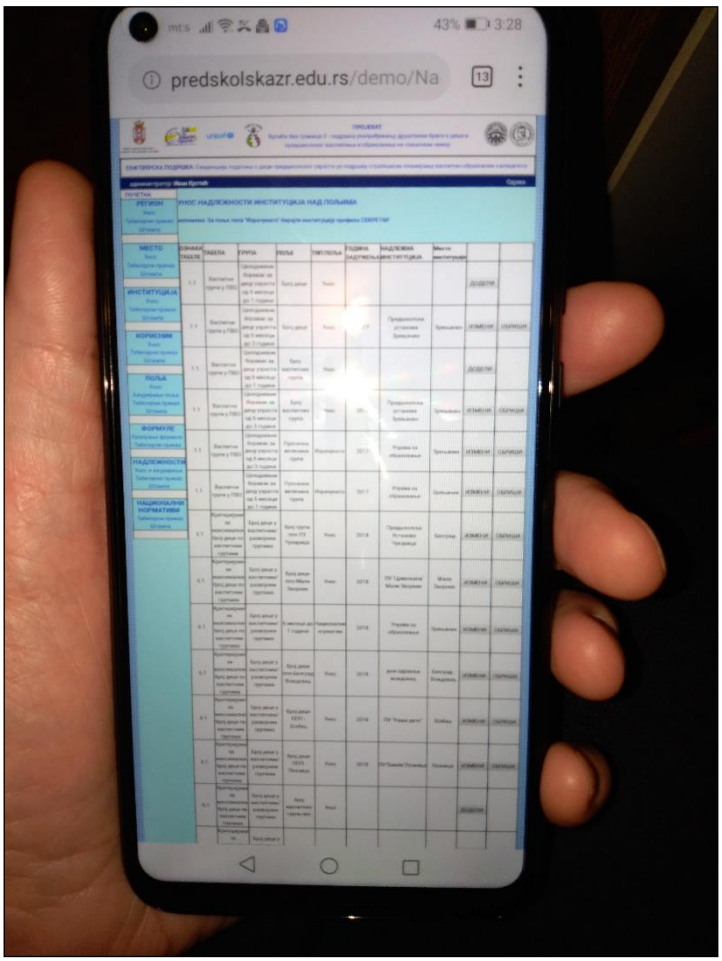


Figure 8. Responsive web design enables presenting the developed web portal at mobile phone

5. ADAPTIVITY OF PRESCHOOL WEB PORTAL

Adaptivity of the developed preschool web portal has many aspects implemented:

I – Adaptivity Dimensions

- *Graphical* – responsive web design that enables web application to automatically fit into any device (desktop monitor, laptop screen, tablet or mobile phone, presented at Figure 8).
- *Functional* – enables the administrator role to define data categories that are to be assigned to any responsible institution (Figure 5), regarding any problem or strategic orientation that needs data entry, integration from any number of institutions and processing. Flexibility in data processing, with options of custom creation of formulas that include raw data, constants (national and local norms) and previously processed data.
- *Platform-based* – software solution could automatically detect the hosting platform (PHP language support version) and adjust key code syntax variant to appropriate platform. The solution consists of:
  - *Sensor* – detecting PHP version supported at hosting (Listing 1),
  - *Effector* – different behavior, i.e. PHP syntax (Listing 2).

```
private function DetectPHPMySQLVersion()
{
        $VersionPHP = phpversion();
        if ($VersionPHP >='5.5.0')
        {
           $this->VersionMYSQLSyntax="mysqli";
        }
        else
        {
          $this->VersionMYSQLSyntax ="mysql";
        }
}
```

Listing 1. Detection of hosting platform PHP version (sensor)

```
if ($this-> VersionMYSQLSyntax =="mysqli")
        {
        $this->connectionDB= mysqli_connect($host,
        $username, $password, $this->databaseName);
        }
        else // mysql
        {
        // establishing connection to DBMS MYSQL
        $this->connectionDBMSMYSQL=
        mysql_connect($host, $username, $password);

        // establishing connection to database
        $this->connectionDB = mysql_select_db(
        $this->databaseName,
        $this->connectionDBMSMYSQL);
        }
```

Listing 2. Different behavior of code, regarding appropriate PHP syntax, according to detected PHP support version at hosting platform (effector)

II – Adaptivity types

- Adaptive software:
  a) Automatically detects hosting PHP support and adjusts php code syntax version;
  b) Automatically adjusts to the user device in responsive web design.
- Adaptable software – functional elements (data categories, institutional data assignments, data processing formulas) are set before the software usage, manually adjusted (by human user in administrative role) to particular problem or strategic goal that is addressed.

6. CONCLUSION

The developed solution of preschool web portal was initially developed for particular strategic goal – to enable data collection, integration and processing related to the problem of detecting the number of children not included in preschools. Having the solution designed more generally, it became a universal tool to support data collection from different institutions, data collection monitoring, data integration, processing, prediction, analysis and visualization. Web portal usage starts with setting a strategic goal and selection of the needed data to be collected regarding reaching the goal. Appropriate data categories are defined and entered in system, as well as assigned to the responsible institutions. Data processing is defined by adding appropriate computing formulas, which combine raw data with national and local norms (constant values).

Particular aim of this paper was to address the issue of making flexible, adjustable software, i.e. web portal. The presented solution shows software adaptivity of adaptive and adaptable type in the aspects: graphical, functional and platform-related. Having all these adjustments integrated in the solution, it brings features such as:

- Scalability to any number of data, users, institutions or processing formulas,
- Universal applicability to any particular problem domain and strategic decision making,
- Automated adaptivity to different hosting platforms and user devices.

This way, it represents one implemented example of adaptive software, which is not only a prototype, but in use. Having all these features, it brings many benefits for using institutions, as well as it makes software maintenance easier.

ACKNOWLEDGEMENTS

This paper presents part of results in the project "Preschool without Frontiers 3 – support to improvement of social care about preschool-aged children at local level". The project was implemented in 2017-2018 year period, by

partners: Ministry of education, science and Technology development of Republic of Serbia, UNICEF, CIP center for interactive pedagogy Belgrade. The web portal was one of the results that was created in year 2018 by team from University of Novi Sad, Technical Faculty "Mihajlo Pupin", Zrenjanin, Serbia in cooperation with Preschool institution Zrenjanin and Center for Interactive pedagogy Belgrade, Serbia.

REFERENCES

[1] Katayama, T (2003). *Science of Software Changes.* Proceedings of the Tenth IEEE Asia-Pacific Software Engineering Conference (APSEC 03).

[2] *Agile Manifesto* (2001) http://www.agilemanifesto.org/ (accessed July 6, 2020).

[3] Salehie M, Tahvildari L (2009). Self-Adaptive Software: Landscape and Research Challenges. *ACM Transactions on Autonomous and Adaptive Systems.*

[4] Zhang J, Cheng B.H.C (2006). *Model-Based Development of Dynamically Adaptive Software.* ICSE'06, Shanghai, China.

[5] McKinley P. K., Sadjadi S. M., Kasten E. P., Cheng B. H. C. (2004), Composing adaptive software*. IEEE Computer,* vol. 37, no. 7, pp. 56–64.

[6] Lehman, M. M. (1996). Laws of software evolution revisited. *Software process technology*. Springer Berlin Heidelberg, 108-124.

[7] Ling Lim, S, Finkelstein, A (2011). Anticipating Change in Requirements Engineering. *Relating Software Requirements and Architectures*, pp. 17-34.

[8] Shi L, Wang Q, Li M (2013). *Learning from evolution history to predict future requirement changes.* IEEE International Conference on Requirements Engineering.

[9] McGee, S, Greer, D (2012). Towards an understanding of the causes and effects of software requirements change: two case studies. *Requirements Eng* 17,133–155.

[10] Derakhshanmannesh, M, Ebert J, Amoui M, Tahvildari L (2011). Introducing Adaptivity to Achieve Longevity for Software. *Lecture Notes in Informatics*, Software Engineering 2011 Workshop, Bonn, pp. 59-70.

[11] Houben, G.J. (2005). *Challenges in Adaptive Web Information Systems: Do Not Forget the Link.* Engineering Advanced Web Applications, Proceedings of Workshops in Connection with 4th International Conference on Web Engineering (ICWE2004), pp. 3-11, Publ. Rinton Press.

[12] Magnaudet, M, Chatty S. (2014). *What should adaptivity mean to interactive software programmers?* EICS 2014, ACM SIGCHI symposium on Engineering interactive computing systems, Jun 2014, Rome, Italy. pp 13-22.

[13] Xiao, L, Greer D (2005). *SOFTWARE ADAPTIVITY THROUGH XML-BASED BUSINESS RULES AND AGENTS*, 17th Int. Conference on Software Engineering and Knowledge Engineering (SEKE'05), Taipei, China, pp. 62-67.

[14] Xiao, L, Greer D (2009). Adaptive Agent Model: Software Adaptivity using an Agent-oriented Model-Driven Architecture. *Information and Software Technology* 51, pp. 109-137.

[15] Manesh M, Ebert J (2011). *Software Evolution Towards Model-Centric Runtime Adaptivity*. Proceedings of the Euromicro Conference on Software Maintenance and Reengineering, CSMR · March 2011.

[16] Ghezzi C, Spoletini P, Tamburelli G (2013). *Managing Non-functional uncertainty via model-driven adaptivity*. Proceedings of International Conference on Software Engineering, May 2013.

[17] Dubey A, Karsai G, Mahadevan N(). Fault-Adaptivity in Hard Real-Time Component-Based Software Systems. In: de Lemos R., Giese H., Müller H.A., Shaw M. (eds) Software Engineering for Self-Adaptive Systems II. *Lecture Notes in Computer Science*, vol 7475. Springer, Berlin, Heidelberg.

[18] Kashi R.N, D'Souza M (2016). *Incorporating Adaptivity using Learning in Avionics Self Adaptive Software: A Case Study*. IEEE Conference on Advances in Computing, Communications and Informatics (ICACCI), Sept. 21-24, 2016, Jaipur, India

[19] Ingolfo S, Silva Souza V.E (2013). *Law and Adaptivity in Requirements Engineering*. IEEE 8th International Workshop Software Engineering for Adaptive and Self-Managing Systems, ICSE Workshops, SEAMS May 2013.

[20] Kazi Lj, Radosav D, Karuovic D, Lojovic T, Kalezic Vignjevic A, Lakicevic O (2019). *Implementing quality aspects of web portal for preschool resources strategic planning. Proceedings of* IX International Symposium Engineering Management and Competitiveness 2019 (EMC 2019), June 21-22, 2019, Zrenjanin, Serbia.

[21] Kazi Lj, Karuović D, Radosav D, Lojović T, Vignjević A.K, Lakićević O (2019). *Design of a web portal for preschool resources strategic planning.* Proceedings of X International conference of Information technology and education development, pp. 93-97.