9th International scientific conference
Technics and Informatics in Education – TIE 2022
16-18 September 2022

Session: *Technics, Technology and Informatics in Education*
Professional paper
DOI: 10.46793/TIE22.049R

Development of information system for digital dialogue in teaching using RESTful service

Momčilo Ranđelović[1*], Miloš Papić[2], Vladimir Veljović[2] and Ljiljana Stanojević[1]
[1]University Business Academy in Novi Sad, Faculty of Social Sciences, Belgrade, Serbia
[2]University of Kragujevac, Faculty of Technical Sciences, Čačak, Serbia
[*] momcilo.randjelovic@fdn.edu.rs

Abstract: *Digital dialogue in teaching, as an asymmetric, multi-user communication system, represents an indirect connection between teachers and students in real time. For this type of communication, it is necessary to ensure the fastest response of clients and to ensure parallel processing as well. The features of RESTful web service, such as: possibility of caching, uniform interface and explicit use of HTTP methods, recommends it as a key component in the information system for digital dialogue in teaching. This paper presents a project for development of such information system using RESTful web service and a description of its advantages compared to other solutions.*

Keywords: *digital dialogue, RESTful, web services, Java, mobile learning*

1. INTRODUCTION

The deployment of mobile devices requires technology that connects mobile systems with a conventional distributed computing environment. If a programmer is writing an application that requires some computation in one place, the write method is computed in the code in that one place. However, if data computation is required in other parts of the application, in distributed systems, it would be impractical to maintain. If a single change is needed in that method, all instances need to be found to edit (and retest).

Such a scenario would benefit from the application of a web service for optimal functioning and improvement of business agility. Web services may be the perfect candidate for such connection, since strong interoperability is a key requirement of this technology. Considering the fact that mobile computing environment is very heterogeneous in terms of hardware platform, operating system and programming language, the integration of mobile computing with web service technology can provide many advantages to both sides [1].

Mobile device technology is in daily development – phones and tablets are becoming computing capable [2], so with appropriate web services they can be equal partners of the web application architecture (they can be a web service client or a web service provider).

Digital dialogue in teaching relies on the use of mobile devices in its key segments [3].

The information system for digital dialogue in teaching contains two functionally different components [4]:

- communication system for dialogue in immediate teaching and
- communication system for DLS support (*Distance Learning System*) (*Figure 1*).

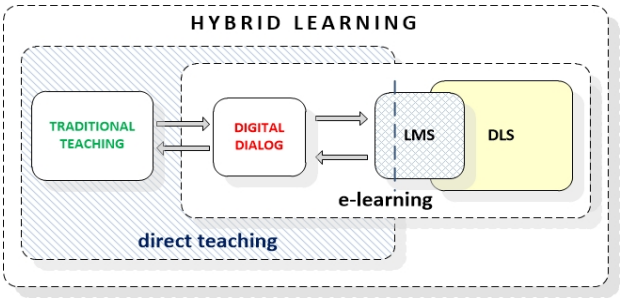

Figure 1: *General scheme for digital dialogue in teaching*

Subjects in the digital dialogue are: student, teacher, parents, professional services, school management and institutions of the Ministry of Education. Each of them can have appropriate, limited access to the system (*Figure 2*).

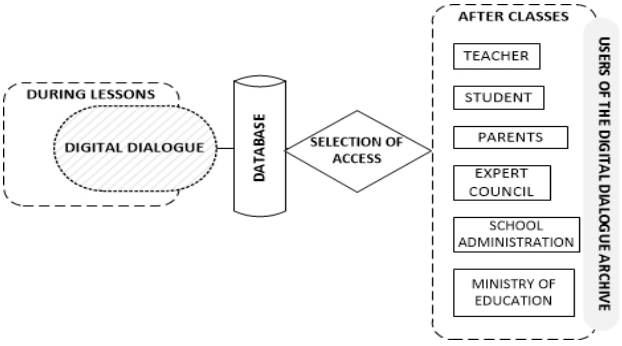

Figure 2: *Users of digital dialogue*

Realization of digital dialogue in teaching requires electronic components for a standard CRS (*Classroom Response System*) environment, which implies an information system that has its own appropriate hardware, system and dedicated application software, database, communication system and data processing methods [4].

The communication system for the implementation of digital dialogue in direct teaching should connect all subjects of the teaching process, through the functioning of the Internet and wireless networks.

The system for connecting students' devices with the teacher's computer can be based on different technologies: *infrared* (IR), *radio-frequency* (RF), SMS, Wi-Fi, LAN. It is necessary to ensure that the broadcast of a large number of responses from students' mobile devices - PRS (*Personal Response System*) in real time is accepted.

The server application accepts client requests and processes them through its interface. The entire system works using web service technology. The application is three-tiered. The upper layer consists of a Windows-form client application for teachers and a web (Android) application for students. These two client applications provide an interface through which data is forwarded to the components. On the middle layer are the components, which enable communication with the database, perform all operations on databases and pass data between layers. They contain most of the system logic. The bottom layer is the data layer.

2. CHOICE OF THE DEVELOPMENT PLATFORM

The server application was implemented in the Eclipse EE (Java Enterprise Edition) IDE 4.24.0 environment in the Java programming language (java.version-17.0.3).

Such an environment already has built-in functionality for handling REST services, which means that there is no need to use an external one or create your own. All persistent data is stored in MySQL database. This approach enables the installation of the entire system within a school or college at the local level by creating a desktop application. Of course, the principle is the same in the case of setting up a web application on the server side (*Figure 3*).

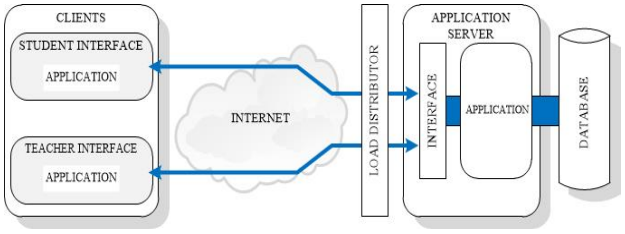

Figure 3: *Communication system for digital dialogue*

2.1. Selection of web services in the application

The choice of web service presents the next step. Web services are basically pieces of software that can be made available over the Internet [4]. They provide an interface between client applications and the server they reside on (*Figure 4*). In essence, this means that web services can be developed in any programming language and then integrated in a relatively simple way [5].

The three main ways of developing web services or APIs (*Application Programming Interface*) are: SOAP (*Simple Object Access Protocol*), XML-RPC (Remote Procedure Call) and REST (*Representational State Transfer*) [6]. A web service has a unique URI (*Uniform Resource Identifier*) that is used to call its function over the Internet.

2.2._SOAP web service as a standard

SOAP is a standardized protocol that enables communication between web services and clients via HTTP (*Hypertext Transport Protocol*) or SMTP (*Simple Mail Transfer Protocol*). Data is transferred using XML (*Extensible Markup Language*). SOAP supports the concept of stateless and one-way message exchange. On the other hand, developers through applications can create significantly more complex interaction patterns, for example request-response, request-multi response, etc.

To use the service, SOAP clients must know the available service directory, the names of the offered operations (*functions*) and the address of the end point for connection. The web services directory itself should be published using the XML-based WSDL (*Web Services Description Language*).

Server applications, developed in Java EE code, SOAP messages can be processed through the JAX-WS API interface (*Java API for XML Web Services*). It is not necessary to use the service directory.

Web services based on the SOAP platform have the following characteristics:

- asynchronous processing
- reliability
- stateful operation – If the application needs additional information, SOAP can offer additional specification in the structure of the web service to support the query (security, transactions, etc.)

There are a number of publicly available SOAP services. The most popular public SOAP services are used on web pages that display weather forecasts, exchange rates, currency conversions, stock prices, etc. Many of them are free, can be connected to the application, but the user is responsible for the appropriate user interface to display the obtained data.

2.3._A RESTful web service as an alternative

REST is a special concept in which domain URLs represent managed objects, and management is realized via HTTP methods. REST is platform and language independent and resistant to Firewall [7]. In fact, the only protocol that REST uses is HTTP. Unlike SOAP and XML-RPC, REST does not require XML Wrappers. This is possible because, as we said, objects are represented by URLs, and behavior is defined by basic HTTP methods: GET, PUT, POST, DELETE.

Responses are also forwarded using the HTTP protocol, but the format itself is not strictly defined. It can be XML, JSON or some other meta language.

The GET method is used to retrieve data, POST to create new data, PUT to update and DELETE to delete data.

Global features of RESTful services are:

- Stateless
- Cacheable
- Uniform interface URI
- Explicit use of HTTP methods
- XMS and/or JSON transfer

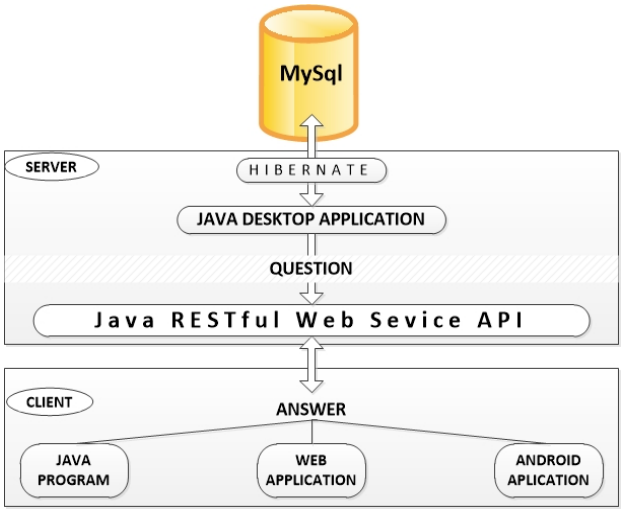

Figure 4: *Information system for digital dialogue for the classroom – architecture*

3. APPLICATIONS IN THE DIGITAL DIALOGUE SYSTEM

3.1._Java desktop application on the server

The desktop Windows application is intended for administrators and teachers. It allows viewing and updating data on system users, both students and other teachers, tasks and tests, viewing statistical data on the use of the application and setting the parameters necessary for the correct operation of the application.

The application itself on the server is composed of two basic modules:

- *Creator* – a module that contains forms for entering data about students, classes and groups, for creating questions with answers, for grouping questions according to teaching units and forms for creating tests.
- *StartDialog* – a module that contains forms for selecting a group of questions or a test and the process of activating the digital dialog itself.

All data in this application is filled in by the teacher. If he knows the curriculum and students well, the teacher can enter most of the data at the beginning of the school year (**Figure *5***).

A set of 5 – 10 standard questions is formed for each class of processing new material. Tests are created for refresher classes (*Figure 6*).

The Creator module in the desktop application does not require a web connection, but a *MySQL* database connection is required.

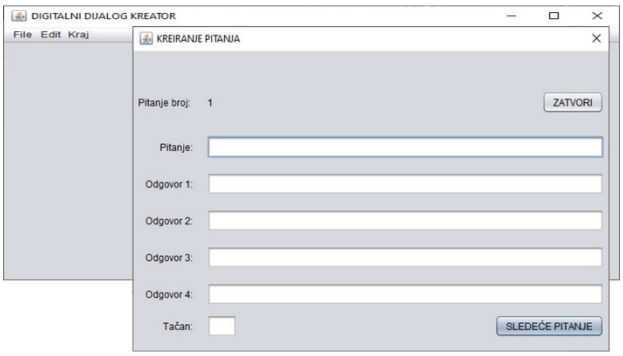

Figure 5: *Form of Java application for entering questions*

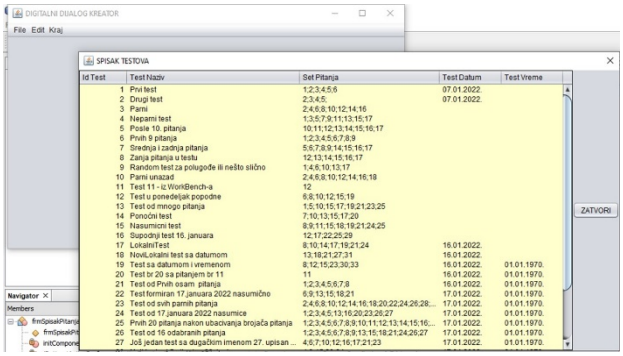

Figure 6: *Form of Java application for creating a group of questions in a teaching unit*

3.2._Java android client side application

On the client side in the digital dialog is an application whose most important role is to provide the user with a display of questions and the possibility of the simplest possible answer.

Since the factor of response speed is also significant, they are most often realized through the form of Multiple Choice Questions (*Figure 7*).

Figure 7: *Client-side Android application*

Unlike the Java application on the server, which is entirely developed in Eclipse IDE, the client application is implemented in the Android Studio Arctic Fox development environment based on REST services [8]. Each client application functions independently, while on the server, after logging in, a new program thread is created for each individual user.

4. STARTING A DIGITAL DIALOGUE

Before the beginning of the class for which the digital dialogue is planned, the teaching topic and the method of communication in the dialogue are chosen (*Figure 8*). Starting the dialogue can be realized in two ways:

- *Student Passed Dialog* – students answer each question successively in their own rhythm, which is mostly not practiced and rarely used in digital dialogue,
- *Teacher Passed Dialog* – the teacher initiates a question and determines the time for providing an answer depending on the current number of received answers, difficulty, type of question, etc.

The second method provides the opportunity for teachers to, at their discretion, allow students a longer or shorter time to provide an answer to each question.

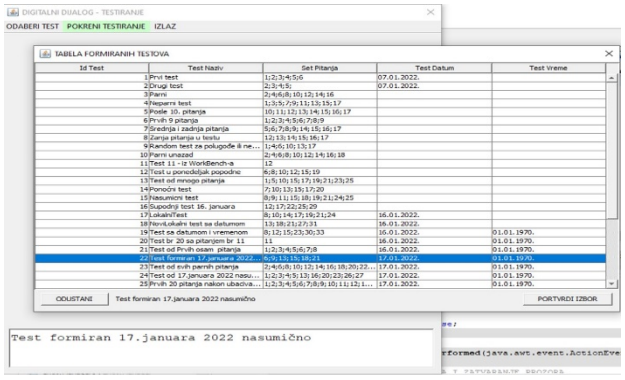

Figure 8: *Form for choosing a test or teaching unit*

During the lesson, the lecturer teaches the lesson using an interactive white board or a projector in the multimedia classroom or computer room.

At the end of each logical unit, according to the teacher's assessment, a question is asked on the board and thus begins the digital dialogue. Students receive a question on their mobile phone, which also appears on the server, with the teacher and have a short period of time to provide an answer (*Figure 9*). The questions are related to the content of the immediately presented material, with the aim that the students notice the most important details of the lecture, keep their attention and check their knowledge. It is also expected that the teacher will receive reliable feedback on the engagement of each student and find out how many students have understood the segment of the teaching material presented.

After each answer, the students wait for a new question, which is also initiated by the teacher.

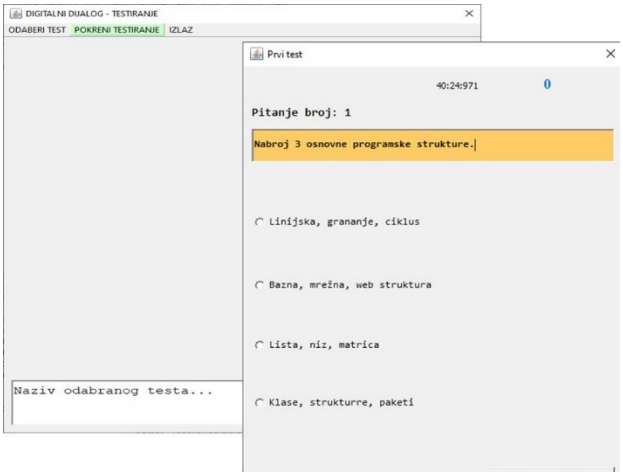

Figure 9: *Window tor displaying questions on the server side*

All questions and all student answers are accepted and automatically recorded in the database. Lesson reports and results of the digital dialogue can be selectively downloaded by students, parents, teachers, professional services or the school director (*Figure 2*), as authorized participants in the digital dialogue. The lesson report includes:

- the total results of all students' answers in the form of a table,
- a report on the activities of each individual student,
- report on individual answers for each question,
- data for analytical purposes, obtained from special database queries.

5. CONCLUSION

After testing, we can conclude that the information system, which uses a RESTful service in its applications, is fully functional. However, the REST service is expected to enable more efficient responses in digital dialogue with a larger number of participants and in multimedia communications.

This could be achieved in university lectures with a large number of students. However, in practice there were not enough opportunities to compare with the previous version, based on SOAP, so these measurements can be done in some future works.

Analyzing the works of other authors [9], [10], [11], [12], [13] and through the process of developing the entire information system for digital dialogue, we came to the following conclusions:

- SOAP and REST cannot be directly compared, since the former is a protocol and the latter is an architectural style,
- REST is protocol independent and not necessarily connected to HTTP,
- REST is more dynamic, no need to create and update UDDI,
- REST is not limited to XML format only,
- RESTful web services can send plain text/JSON/XML,
- SOAP is more standardized and new, greater security,
- The SOAP client works as a custom desktop application, tightly coupled to the server,
- by using SOAP, constant updates are required after each change, but it is easier to determine whether the contract is being followed,
- the client can use the REST service without knowing the API, only the entry point and the media type,
- in SOAP, the client needs prior knowledge of everything it will use, or it won't even start the interaction,
- The REST client can be extended with code on request delivered by the server itself,
- A REST application can use any protocol for which there is a standardized URI scheme,
- REST is more difficult to develop at first, but pays off over time with easier evolution on the server side and client resistance to changes

AKNOWLEDGMENTS

This study was supported by the Ministry of Education, Science and Technological Development of the Republic of Serbia, and these results are parts of the Grant No. 451-03-68/2022-14/200132 with University of Kragujevac – Faculty of Technical Sciences Čačak.

REFERENCES

[1] Lj. Stanojević and B. Rakić, "Mobile Technology in Higher Education – A Students Perspective," in *Seventh International Scientific Conference Employment, Education And Entrepreneurship (EEE2018)*, Belgrade, 2018.

[2] S. Minić, D. Kreculj and M. Vorkapić, "Upotreba i značaj WLAN mobilnih tehnologija u nastavi," in *Technology Informatics and Education for Learning and Knowledge Society - 6th International Symposium*, Čačak, 2011.

[3] M. Ranđelović, "Primena informacionih tehnologija u realizaciji nacionalnog takmičenja - Tesla Info Kup," Elektronski fakultet Univerziteta u Nišu, Niš, 2013.

[4] M. Ranđelović, A. Veljović, L. Stanojevic and L. Paunović, "Efekti digitalnog dijaloga u nastavi programiranja," in *Technics and Informatics in Education, 6th International Conference*, Čačak, 2016.

[5] D. Rathod, "Performance evaluation of restful web services and soap / wsdl web services," *International Journal of Advanced Research in Computer Science,* pp. 415-420, 2017.

[6] P. Yu-Yen, M. Shang-Pin and L. Jonathan, "REST2SOAP: A framework to integrate SOAP services and RESTful services," in *IEEE International Conference on Service-Oriented Computing and Applications (SOCA)*, 2009.

[7] B. Abhijit and B. Tulshi, "A Comparative Investigation on Implementation of RESTful versus SOAP based Web Services," *International Journal of Database Theory and Application,* vol. 8, pp. 297-312, 2015.

[8] Ei Ei Thu and Than Nwe Aung, "Developing Mobile Application Framework by Using RESTFul Web Service with JSON Parser," in *International Conference on Genetic and Evolutionary Computing*, Mandalay, Myanmar, 2015.

[9] S. Mumbaikar and P. Padiya, "Web Services Based On SOAP and REST Principles," *International Journal of Scientific and Research Publications,* pp. 1-4, May 2013.

[10] R. V. V. S. B. Ricardo Ramos de Oliveira, "Comparative Evaluation of the Maintainability of RESTful and SOAP-WSDL Web Services," in *IEEE International Symposium on the Maintenance and Evolution of Service-Oriented and Cloud-Based Systems (MESOCA)*, Eindhoven, 2013.

[11] R. Daigneau, Service Design Patterns - Fundamental Design Solutions for SOAP/WSDL and RESTful Web Services, Westford, Massachusetts: Pearson Education, 2012.

[12] F. Alshahwan and K. Moessner, "Providing SOAP Web Services and RESTful Web Services from Mobile Hosts," in *Internet and Web Applications and Services, International Conference on*, Barcelona, 2010.

[13] J. Tihomirovs and J. Grabis, "Comparison of SOAP and REST Based Web," in *Information Technology and Management Science*, Riga, 2016.