

# DEVELOPMENT OF WEB SERVICES BASED ON XML LANGUAGE AND CONCERNING THE SOAP PROTOCOL

<sup>1</sup>VLADIMIR SIMOVIC, <sup>2</sup>MATIJA VARGA, <sup>3</sup>IVAN CVITKOVIC

<sup>1</sup>Algebra University College Zagreb & University of Applied Sciences Baltazar Zapresic (Croatia-EU)

<sup>2</sup>University North Koprivnica & University of Applied Sciences Baltazar Zapresic (Croatia-EU)

<sup>3</sup>University North Koprivnica (Croatia-EU)

E-mail: <sup>1</sup>vladimir.simovic@algebra.hr, <sup>2</sup>mvarga@unin.hr, <sup>3</sup>icvitkovic@unin.hr

---

**Abstract** - This research paper on the topic: of "Development of WEB Services Based on XML Language and Concerning the SOAP Protocol" will explore: what is the main purpose of the SOAP protocol, its shortcomings of the SOAP protocol, whether there are security vulnerabilities within the SOAP protocol and security options. In addition to the research part, the paper will also use a scientific method of content analysis based on which certain definitions of WEB services will be given, such as (1) XML, (2) AJAX, (3) SOAP and (4) REST. The research will be conducted on respondents (sample) belonging to the group of web developers (in terms of collecting relevant information). Also, in theory, the following will be explained in detail: (1) web services and (2) protocols such as (1) AJAX protocols, (2) SOAP protocols, and (3) REST state transfers. In addition to the mentioned protocols and XML as an extensible language for tagging data and documents, formats such as (1) HTML 5.3 and (2) JSON open standards for formatting data when transferring between applications are also an indispensable part of the web service, as for the scientific practice of "FlexSim" 3D Simulation Modelling and Analysis Software usage (<https://www.flexsim.com/>) with aims to analyse and assess the possibility of applying intelligent transport systems, i.e. advanced digital technologies in road traffic to increase traffic flow, safety and comfort.

---

**Keywords** - AJAX, HTML & XML, JSON & REST, SOAP & Web Services

---

## I. INTRODUCTION

This research paper on the topic: "Development of WEB Services Based on XML Language and Concerning the SOAP Protocol", it was investigated: what is the purpose of the SOAP protocol, the flaws of the SOAP protocol, the existence of security flaws within the SOAP protocol and the possibilities of protection, and the functioning model of the SOAP protocol. In addition to the research part, the paper also used the scientific method of content analysis, based on which certain definitions of WEB services were defined, such as (1) XML, (2) AJAX, (3) SOAP and (4) REST, and the scientific method of modelling, which modelled the principles of work. SOAP protocol and SOAP message structure. The research was conducted with respondents (sample) who belong to the group of web developers (in terms of collecting relevant information). Also, theoretically explained in detail: (1) web services and (2) protocols such as (1) AJAX protocol, (2) SOAP protocol and (3) REST state transfer. In addition to the mentioned protocols and XML as an extensible language for marking data and documents, an indispensable part of web services are formats such as (1) HTML 5.3 and (2) JSON, an open standard for formatting data when transferring between applications, which are also included in the research work, as the scientific practice of "FlexSim" 3D Simulation Modelling and Analysis Software usage (<https://www.flexsim.com/>) with aims to analyse and assess the possibility of applying intelligent transport systems, i.e. advanced digital technologies in road traffic to increase traffic flow, safety and comfort. JSON is a JavaScript Object Notation or JSON (JavaScript Object Notation), i.e.,

JSON is an open standard for formatting data when transferred between applications and has a human-readable syntax. JSON stores data in the form of attributes and values, and can also have a field as a data type. JSON came out of JavaScript and many programming languages support it. With web services, the protocols used are of great importance because they tell how data is transferred and how the application communicates over the Internet. Some of the more important and popular protocols are (1) AJAX, (2) SOAP, and (3) REST.

## II. SOAP PROTOCOL AND ITS PURPOSE

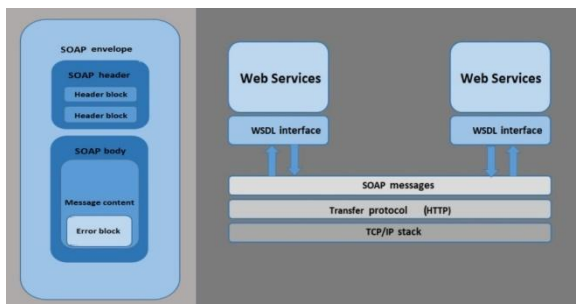
### A. The SOAP Protocol

The SOAP protocol is a basic communication protocol intended for the exchange of text messages with web services. It describes how the message will be formatted during transmission to some of the transport protocols and how the same message will be exchanged and processed between applications [1]. The language in which SOAP messages are written is called XML (Extensible Markup Language), which stands for extensible description language.

### B. The Working Principle of the SOAP Protocol and the Structure of the SOAP Message

Figure 1 shows the working principle of the SOAP protocol and the structure of the SOAP message. Each SOAP message that is sent by an application, service, or server that uses a particular web service is transported by a network transport protocol. In this case, the protocol is HTTP (Figure 1), through which the request is sent to the server. In addition to HTTP, SOAP uses the FTP (File Transfer Protocol) protocol,

which enables the transfer of files from the client to a remote computer, i.e., an FTP server. Also, SOAP defines a standard communication protocol based on the exchange of XML messages. It uses different transport protocols such as HTTP, JMS and SMTP. The standard HTTP protocol makes it easy for the SOAP model to pass through firewalls and proxy servers without additional modifications. The SOAP client, by the SOAP specification, creates an XML document containing the corresponding request. That document arrives at the SOAP server, the server processes the incoming requests, and upon completion, using the SOAP protocol returns a response message to the SOAP client. Upon receiving the request, the web server opens a WSDL document containing instructions for processing the request, and then the web service processes the SOAP message and displays it to the user using an interface or as a text record. After the web service has received the message with the request and processed the message, the processing result is sent to the application that decides what kind of response and how many responses it sends back to the user or the web service that originally sent the request.



**Figure 1: The working principle of the SOAP protocol and the structure of the SOAP message**  
 (Source: Created by the authors of the paper)

Figure 1 also shows how a SOAP message consists of a SOAP protocol envelope. It is written in the XML language for creating databases at the application level, and XML is used in most web applications. It is a very simple concept similar to the HTML description language, consisting of tags that are keywords in a SOAP message. A SOAP envelope (Figure 1) contains two basic elements: (1) a SOAP header and (2) a SOAP body. The SOAP header contains the header blocks while the SOAP body contains the message content and the error block. The interface based on Figure 1 can be created using the WSDL language, which is very similar to HTML in the way tags are written, and is very understandable and easy to write [2]. WSDL is a language that describes the behaviour and actions of a web service according to a SOAP message and vice versa. The full name of the abbreviation WSDL is in English Web Service Description Language. It sets abstract rules according to which each web service processes incoming messages and sends responses to messages.

### C. View the Source Code That Defines a SOAP Message

Based on Figure 1, the structure of the part of the source code that defines the SOAP message looks like this in Figure 2.

```

1  <soap:Envelope>
2    <soap:Header>
3
4    <!-- SOAP ZAGLAVLJE -->
5
6    </soap:Header>
7    <soap:Body>
8
9    <!-- SOAP TIJELO -->
10
11   </soap:Body>
12 </soap:Envelope>
13
14
15 <?xml version="1.0"?>
16 <soap:Envelope
17   xmlns:soap="http://www.w3.org/2003/05/soap-envelope/"
18   soap:encodingStyle="http://www.w3.org/2003/05/soap-encoding">
19   <soap:Header>
20     ...
21   </soap:Header>
22   <soap:Body>
23     ...
24     <soap:Fault>
25       ...
26     </soap:Fault>
27   </soap:Body>
28 </soap:Envelope>
    
```

**Figure 2: View the Source Code That Defines a SOAP Message**  
 (Source: Created by the authors of the paper)

Figure 2 shows the source code that defines a SOAP message based on the model from Figure 1. Also, the image shows, for example, the source code of the w3schools page that defines a SOAP message [5].

### III. SECURITY OF SOAP WEB SERVICES

The standard HTTP protocol makes it easier for the SOAP model to pass through firewalls and proxy servers without additional modifications. For the connection to the SOAP web service to be secure, it is necessary to set the URL of the service to HTTPS so that all data is transmitted via a secure layer, i.e., SSL (Secure Socket Layer). SSL is a protocol that enables HTTPS and relies on asymmetric encryption. In asymmetric cryptographic systems, there are two types of cryptographic keys - public and secret cryptographic keys. We use the public cryptographic key exclusively for encryption and the secret key for decryption.

The public and secret cryptographic keys form a unique pair; each public key is accompanied by a unique secret key. In practice, it is very difficult, almost impossible, to know one of them to calculate the other. It works in such a way that the client, in this case, the application that sends the requests, obtains a public key via an SSL certificate and uses this to initiate secure communication with the web service. While, on the other hand, the web service keeps its private, key secret and uses it to decrypt received requests. The SOAP protocol enables the communication between applications running on different operating systems and different technologies.

SOAP is not the best choice for implementing communication for the following reasons:

- enhanced security provided by SOAP is not necessary in some cases,
- the entire architecture of individual information systems is located within the organization's infrastructure, where existing security protocols and procedures are used,
- existence of a rigid "relationship" between SOAP client and server,
- when either party changes, the "relationship" is broken and communication stops working,
- XML, which is used to create SOAP messages, can become very complex and difficult to maintain, and thus error-prone and demanding to parse,
- most of the data of the SOAP XML message builds the structure of the SOAP envelope (Figure 1) of the message and represents the overhead that must be sent with each message,
- The SOAP service requires maintaining an open connection to the client, and testing requires a ready-made SOAP client or a separate software tool [3].

With web services, the protocols used are of great importance because they tell how data is transferred and how the application communicates over the Internet. Some of the more important and popular protocols are (1) AJAX, (2) SOAP, and (3) REST. It should also be noted that AJAX applications (when it comes to security) have increased exposure to attacks (exposed surface) due to the transparency of communication between the client and the server, which attackers can use to gather information about the implemented mechanisms of the web application.

#### IV. THE ROLE OF AJAX AND REST

The role of REST (Representational state transfer) and its advantages compared to the SOAP protocol are: (1) faster responses to requests, (2) lower implementation costs, (3) simpler maintenance of server-side communication, and (4) less memory usage [4]. The REST protocol can use many different responses (XML, JSON, etc.). REST services provide good temporary stores (cache) when using GET calls [4].

Currently, the best solution for communication between technologies is the REST (Representational state transfer) protocol. REST took great features from the SOAP protocol, such as neutrality, i.e., the ability of programming languages such as C/C++, C#, Java and similar programming languages, to communicate independently of the operating system (MS Windows family, Linux, etc.) [4]. REST was created from WWW (World Wide Web) technology, i.e., a service, by introducing certain restrictions. These restrictions form the basic principles of the REST model (Figure 4) that determine how resources on the global Internet can be used.

The motivation for introducing these restrictions is to create a final system that takes advantage of all the benefits of the web architecture to make the system work better. Since the aforementioned non-standard web services often implement a part of the theoretical principles of the REST model, the name RESTful web service(s) is used for that (Figure 4). This emphasizes that some REST principles are used, but not all REST principles.

```

1 <user UserId=345>
2   <Name>Matija</Name>
3   <Lastname>Varga</Lastname>
4   <Locatio>Zagreb</Locatio>
5   <Faculty>North University</Faculty>
6   <Status>OK</Status>
7 </user>
    
```

Figure 3: Example of a simple REST request and response written in XML

(Source: Created by the authors of the paper)

Figure 3 shows an example of a simple REST request and response written in XML. The basic system elements of the REST model are called resources. Each web service offers users access to a finite number of resources.

For example, popular web services, such as the Amazon API, allow retrieving book titles, book authors, critic reviews, and other information. Books and reviews are resources that the web service offers to the user. URI (Uniform Resource Identifier) was chosen as the unique resource identifier. In this way, the resource is assigned a unique identifier within the global computer network of the Internet.

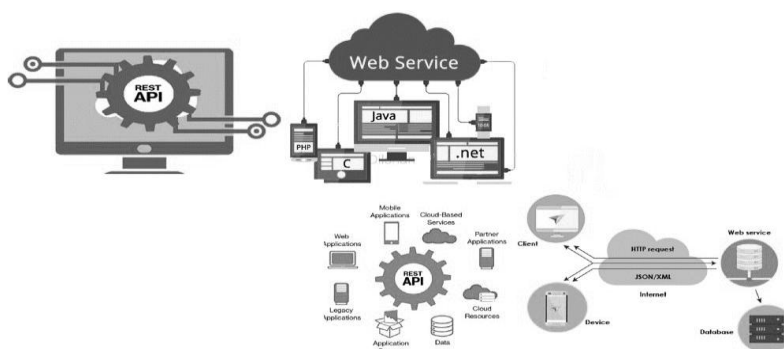


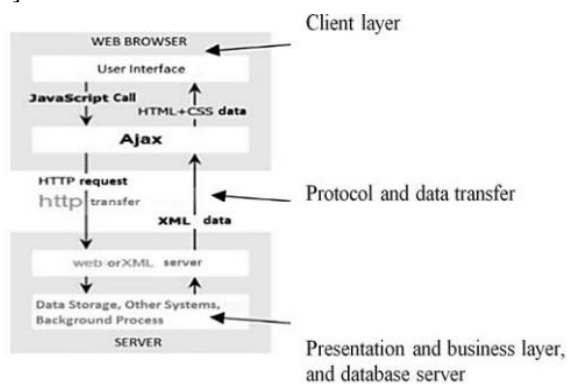
Figure 4: REST Ful Web Service structure

(Source: Created by the authors of the paper based on the source: [6])

Figure 4 shows RESTful Web services that include applications created in programming languages such as Java, PHP, C programming language and \*.net. In addition to the REST principle that each resource has a unique identifier, there are other principles such as (1) interconnection of resources, (2) use of standard methods, (3) resources with multiple representations and (4) communication without maintaining state. Asynchronous JavaScript and XML or AJAX is a set of related technologies for developing web applications. The use of the mentioned development techniques increases interactivity on web pages, and CSS is most often used for presentation. It is possible to use them with server technologies, but most often everything takes place client-side, without the need to reload data and additional transmission from the server. JavaScript is a simple, interpreted scripting language intended primarily for the development of interactive HTML pages. The JavaScript core is included in most browsers today such as Internet Explorer, Google Chrome, Mozilla Firefox, Opera, Safari and others). JavaScript makes it possible to perform certain actions in otherwise static HTML documents, such as interacting with the user, changing properties of window browsers, or dynamically creating HTML content. JavaScript is not a simplified version of the Java programming language. They are connected only by similar syntax and the fact that they are used to perform certain actions within the browser [7].

## V. CONCLUSION

Through the research on the topic: " Development of WEB Services Based on XML Language and Concerning the SOAP Protocol", certain definitions of WEB services such as (1) XML, (2) AJAX, (3) SOAP and (4) REST were defined, and a scientific modelling method was developed. model of the working principles of the SOAP protocol and the structure of the SOAP message (Figure 1 and Figure 4), and according to research (interviews) conducted on a group of respondents (about forty experts with experience in the scientific and professional field that was researched, and they are also web developers (in terms of collecting relevant information).The primary reason why AJAX technologies are explored in this paper, among other things, is that AJAX technology enables access to web applications at a speed that is also present when accessing common desktop applications on a personal computer (which is certainly a significant innovation). A complete RESTful Web Service model is presented using a scientific modelling method, which shows in detail which applications and programming languages and protocols are covered. The paper also states the principles of REST web services. Also, the purpose of this work is to help in the future technically supported and automated analysis and assessment of the possibility of applying intelligent transport systems or advanced digital technologies in road traffic to increase traffic flow, safety and comfort. Future research aims to make an excellent analysis and assessment of the position of influence of one of the modules of intelligent transport systems, the time counter of signal terms on approaches to traffic-lighted intersections, based on scientifically based research methodology. Also, the desire is to demonstrate their influence on the quality of traffic service on the examples of traffic flow analysis with and without counters, which could be more easily automated and analysed in the future through the development of WEB Services Based on XML Language and Concerning the SOAP Protocol.



**Figure 5: Presentation of the functioning of applications that use AJAX technologies**  
 (Source: Created by the authors of the paper based on the source: [7])

Figure 5, adapted from the source [7], shows the functioning of applications that use AJAX technologies. Figure 5 shows the client layer, which includes: (1) the web browser, (2) the user interface it uses, (3) calling JavaScript, (4) AJAX technology, and (5) HTML5.3 and CSS3 data. With the AJAX technology from Figure 5, the HTTP request protocol and the HTTP transfer protocol were used. The server contains the XML database and Data Storage, other systems, and numerous other processes that take place in the background.

## REFERENCE

- [1] W3C. URL: <https://www.w3.org/TR/soap12/> (accessed10/15/2022)
- [2] W3C. Web Services Description Language (WSDL) Version 2.0 Part 1: Core Language. URL: <https://www.w3.org/TR/2007/REC-wsdl20-20070626/> (accessed10/15/2022)
- [3] M. Matejas, "Adaptable model of integration of the management of contracting services in banking with the existing system", Doctoral thesis, University of Zagreb, Faculty of Electrical Engineering and Computing, 2019. URL: <https://repozitorij.fer.unizg.hr/islandora/object/fer%3A6615/d/atastream/PDF/view> (accessed10/16/2022)
- [4] A.Simec, and D.Lozić, "Smart communication on the Internet via the REST protocol", Mipro 2014, Ed. P. Biljanovic, Opatija, pp. 1498-1505, 2014.

- http://iraj.in
- [5] Skeleton SOAP Message. URL: [https://www.w3schools.com/xml/xml\\_soap.asp](https://www.w3schools.com/xml/xml_soap.asp).(accessed10/16/2022)
- [6] D. Ramesh, RESTful Web services. URL: <https://medium.com/@dilshanramesh81/restful-web-services-35a96401b42b>. (10/17/2022)
- [7] V. Simovic, M. Varga, and P. Oreski. "Standard Languages for Creating a Database to Display Financial Statements on a Web Application", World Academy of Science, Engineering and Technology International Journal of Computer, Electrical, Automation, Control and Information Engineering, vol. 11, no. 2,pp. 174-177, 2017. URL: <https://publications.waset.org/10006315/standard-languages-for-creating-a-database-to-display-financial-statements-on-a-web-application>.(accessed10/17/2022)

★ ★ ★