|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| VILNIAUS KOLEGIJA  UNIVERSITY OF APPLIED SCIENCES  FACULTY OF ELECTRONICS AND INFORMATICS  Image result for viko logo | | |  | | | VILNIUS COLLEGE  Image result for viko logoFACULTY OF ELECTRONICS AND INFORMATICS |
|  | | |  | | |  |
| **WEB SERVICES** | | |  | | | **INTRODUCTION TO INFORMATICS** |
| PRACTICAL ASSIGNMENT  E-SHOP CREATION WITH CMS USING WEB SERVICES  6531BX028 PI18E | | |  | | | PRACTICAL ASSIGNMENT  SPOTIFY USER MANUAL  6531BX028 PI18E |
| STUDENT | DŽIUGAS PEČIULEVIČIUS | STUDENT | | DŽIUGAS PEČIULEVIČIUS |
| (SIGNATURE)  4/8/2020 | | |  | | | LECTURER |
| LECTURER | AIRINA SAVICKAITĖ | (SIGNATURE)  10/17/2018 | | VIRGILIJUS KUKLIERIUS |
| (SIGNATURE)  4/8/2020 | | |  | | | 2018 |

2020

**TABLE OF CONTENTS**

[INTRODUCTION 4](#_Toc41861404)

[PURPOSE 5](#_Toc41861405)

[1. E-COMMERCE AND CMS 6](#_Toc41861406)

[**1.1.** **What is E-commerce and CMS?** 6](#_Toc41861407)

[**1.2.** **Web services in ecommerce** 7](#_Toc41861408)

[2. EXTENSIVE MARKUP LANGUAGE - XML 9](#_Toc41861409)

[**2.1.** **What is Extensive Markup Language?** 9](#_Toc41861410)

[**2.2.** **XML Sitemap** 9](#_Toc41861411)

[**2.3.** **Advantages of XML towards E-Commerce** 9](#_Toc41861412)

[3. SIMPLE OBJECT ACCESS PROTOCOL - SOAP 11](#_Toc41861413)

[**3.1.** **What is Simple Object Access Protocol?** 11](#_Toc41861414)

[**3.2.** **Advantages of SOAP towards E-Commerce** 11](#_Toc41861415)

[4. WEB SERVICES DESCRIPTION LANGUAGE - WSDL 12](#_Toc41861416)

[**4.1.** **What is Web Services Description Language?** 12](#_Toc41861417)

[5. UNIVERSAL DESCRIPTION DISCOVERY AND INTEGRATION - UDDI 13](#_Toc41861418)

[**5.1.** **What is Universal Description Discovery and Integration?** 13](#_Toc41861419)

[**5.2.** **Advantages of UDDI towards E-Commerce** 14](#_Toc41861420)

[CONCLUSIONS 15](#_Toc41861421)

[LITERATURE 16](#_Toc41861422)

[SOAP & WSDL WORDPRESS PRACTICE 18](#_Toc41861423)

[**1.** **Authentication token** 19](#_Toc41861424)

[**2.** **Getting a post** 22](#_Toc41861425)

[**3.** **Getting a user** 23](#_Toc41861426)

[**4.** **Getting an option** 24](#_Toc41861427)

[**5.** **Getting a taxonomy** 25](#_Toc41861428)

[**6.** **Getting a term** 27](#_Toc41861429)

**TABLE OF FIGURES**

[Figure 1 - Web services technologies and their roles 7](#_Toc41861380)

[Figure 2 - Simple WSDL example 12](#_Toc41861381)

[Figure 3 - Web Services-Based Architecture 13](#_Toc41861382)

[Figure 4 - SOAP API 18](#_Toc41861383)

[Figure 5 - WSDL message 1 19](#_Toc41861384)

[Figure 6 - WSDL message 2 20](#_Toc41861385)

[Figure 7 - SOAP Server 20](#_Toc41861386)

[Figure 8 - SOAP SERVER LOCATION 20](#_Toc41861387)

[Figure 9 - authentication form 21](#_Toc41861388)

[Figure 10 - authentication call 21](#_Toc41861389)

[Figure 11 - authentication response 21](#_Toc41861390)

[Figure 12 - get post form 22](#_Toc41861391)

[Figure 13 - get post call 22](#_Toc41861392)

[Figure 14 - WP post link 22](#_Toc41861393)

[Figure 15 - get user form 23](#_Toc41861394)

[Figure 16 - get user call and response 23](#_Toc41861395)

[Figure 17 - get option form 24](#_Toc41861396)

[Figure 18 - get option call and response 24](#_Toc41861397)

[Figure 19 - get taxonomy form 25](#_Toc41861398)

[Figure 20 - created taxonomy 25](#_Toc41861399)

[Figure 21 - get taxonomy call and response 26](#_Toc41861400)

[Figure 22 - get term form 27](#_Toc41861401)

[Figure 23 - getting term\_id 27](#_Toc41861402)

[Figure 24 - getTerm call and response 28](#_Toc41861403)

# INTRODUCTION

A content management system (CMS) is a term used to describe a system that manages Web site content. The CMS is well known for its ability to provide templates that allow a content author to manage the creation, modification and removal of Web site content without the need to know any of Hypertext Markup Language (HTML) or any other website building language knowledge. This way users or even organizations are less dependent on specialized web developer skills for website development and maintenance. Maintenance is also less complicated because developers only need to modify templates instead of individual webpages. More and more, companies wanted to customize Websites to appeal to different audiences, particularly in sites for electronic commerce.

# PURPOSE

The goal for this assignment are:

* To get familiar with electronic shopping (e-Commerce) platforms and explain what content management systems (CMS) are.
* Have a look at web services that are used in building e-commerce applications.
* How web services help building an e-commerce website?
* What features web services provide?

# E-COMMERCE AND CMS

## **What is E-commerce and CMS?**

First of all, what is E-commerce?

Ecommerce is known as electronic or internet commerce. It refers to the buying and selling goods or services using the internet, and the transfer of money and data to execute these transactions. Ecommerce is often used to refer to the sale of physical products online, but it can also describe any kind of commercial transaction that is facilitated through the internet. Whereas e-business refers to all aspects of operating an online business, ecommerce refers specifically to the transaction of goods and services [1].

Basically the e-commerce is all about shopping online and transfering money from one end to another. There are six types of e-commerce:

* Business-to-Business (B2B)
* Business-to-Consumer (B2C)
* Consumer-to-Consumer (C2C)
* Consumer-to-Business (C2B)
* Business-to-Administration (B2A)
* Consumer-to-Administration (C2A)

Now what exactly is Content Management System (CMS)?

CMSs are web development tools, enabling users to create, edit and store web content. Form most CMSs, this can be done without much or any knowledge of how to code. Plugins or themes are available for many CMSs which increase the functionality and provide a variety of pre-coded design options. Content quality products in a convenient and efficient way [2].

There are many Content Management Systems available (WordPress, Joomla, Drupal, Shopify and many more), some are edited for specific functionality such as selling products and services online. CMS is a software that is used to create and manage content on websites.

## **Web services in ecommerce**

Web services simplify application design, provide reusability and flexibility, deliver enhancements more quickly and reuse software more easily. Web services provide businesses with a standard way to communicate. Retailers are turning to Web services, which rely on Web-based, distributed computing and program-to-program communications.

The main idea in Web services is the **Service-Oriented Architecture (SOA),** a coupled mechanism for software providers to make modularized functionality accessible over a network [5].

Web services replace complex, proprietary programming interfaces with **Extensive Markup Language (XML)** documents and standard application protocols. XML is used to tag data, and **SOAP or REST** is used to transfer it. **Web Services Description Language (WSDL)** describes the services needed and **Universal Description Discovery and Integration (UDDI)** tells applications what services are available.

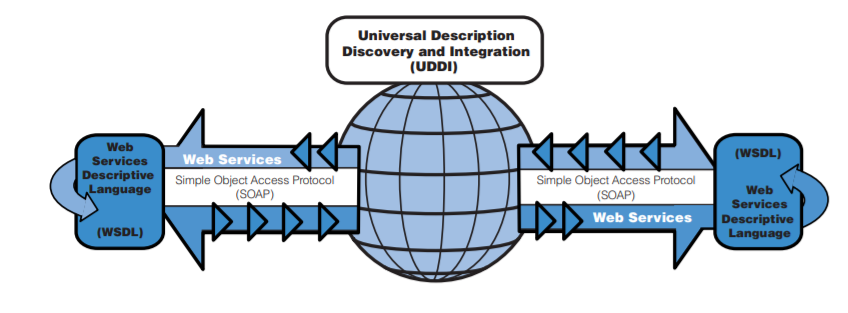


Figure 1 - Web services technologies and their roles

Web services also simplify credit card transactions, in that case businesses an exchange data with their suppliers, partners and customers more efficiently. A web service could carry out payment validation by taking information from one system and initiating a request for more information relevant to the process. For example, a customer provides their name and address and later they need to fill their credit card information.

Amazon has been in the web services business since it was launched in 2002. It’s called Amazon Web Services (AWS). The service provided software developers, web site owners and merchants with access to back-end features found on Amazon’s web site, such as payment systems.

# EXTENSIVE MARKUP LANGUAGE - XML

## **What is Extensive Markup Language?**

Extensive Markup Language (XML) provides us with a universal data exchange standard that allows access to data irrespective of its format and location. XML is basically a markup language used to describe the data, with a strong descriptive, extensible, cross-platform feature.

Web services are XML-centered data exchange systems that use the internet for A2A (application-to-application) communication and interfacing. These processes involve programs, messages, documents and objects [16].

A key feature of web services is that applications can be written in various languages and are still able to communicate by exchanging data with one another via a web service between clients and servers. A client summons a web service by sending a request via XML, and the service then responses with an XML response [16].

## **XML Sitemap**

A XML sitemap is a page which is not visible to a user, but is very useful for search engines. A sitemap is a file where a developer or a person who is creating it provides information about the pages, videos and other content on the website and relationships between the files. A sitemap tells google which pages and files are more important and provide a valuable information about these files like when it was last or how frequently it gets updated.

## **Advantages of XML towards E-Commerce**

XML has some advantages to both the business and its customer.

From a business perspective, XML offers:

* **Standardization.** It’s crucial to B2B (business to business) and B2C (business to customer) e-commerce. XML is platform and application independent [7].
* **Manageability.** It is extremely easy to manage and transform to different types of outputs for media devices if needed without needing to modify the original content [7].
* **Longevity.** XML data exists as plain text. This gives data has a longer life span with future readability and reuse of data [7].

From a customer’s perspective, XML offers:

* **Personalization.** There are many ways of accessing information. XML improves the presentation of the information for its end use because it separates the structure of data from the way data is presented. When marked in an XML format, agents can customize the information to be viewed according to the preferences of the end user and user’s device [7].
* **Accessibility.** Web is increasingly being accessed by customers using a variety of devices with differences in device capabilities. XML being a platform-independent, scales well to provide information on these devices without the need for the user to make adjustments at his/her end [7].

# SIMPLE OBJECT ACCESS PROTOCOL - SOAP

## **What is Simple Object Access Protocol?**

SOAP is a Simple Object Access Protocol. This protocol is for the exchange and transfer of structured info in computer networks and web services. SOAP makes use of XML for the message format. It also uses other protocols, such as HTTP [8].

A SOAP message consists of an “Envelope”, an optional “Header”, and a mandatory “Body”. The SOAP “Body” carries application-specific contents including the method name and the serialized values of the methods’ input or output parameters. Parameters of a Web services method can be a simple value or a compound value (structure or array). Serializing a web services message in XML format allows the SOAP XML to pass through internet firewall [9].

To sum up how what is SOAP web services – it’s defined as Simple Object Access Protocol. This web service protocol exchanges structured data using XML and generally HTTP for transmission. SOAP also uses WSDL documents to distribute a web service description model. This describes how the SOAP requests (client-side) and responses (server-side) must appear [16].

## **Advantages of SOAP towards E-Commerce**

SOAP protocol offers some advantages for business. It is a platform independent protocol, and it works with the XML language. It is a simple way to use and interact with remote service communications, components, objects and more. When compared with other protocols, most people find that SOAP is very easy to learn for anyone who already has understanding of XML [8].

SOAP is a lightweight protocol that uses XML. SOAP is thus not dependent on a specific vendor and therefore the business will not get tied in to specific vendor or package.

Interfaces are clearly defined in such way that they are easy to access and understand. It formalizes the definition of the required interfaces, essentially enforcing a contract between the caller and the receiver. This is prime importance when connecting to distributed applications.

SOAP can also function over standard HTTP where other distributed systems may run into the firewall. SOAP makes use of current internet infrastructures.

# WEB SERVICES DESCRIPTION LANGUAGE - WSDL

## **What is Web Services Description Language?**

Maintained by W3C just like XML (World wide web consortium) is s an XML-based format for describing Web services. Clients wishing to access a web service can read and interpret its WSDL file to learn about the location of the service and its available operations. In this way, the WSDL definition acts as the initial Web service interface, providing clients with all the information they need to interact with the service can be accessed, what operations the service performs, the communication protocols the service supports and the correct format for sending messages to the service.

A WSDL file is an XML document that describes a Web service using 6 elements:

* Port type – groups and describes the operations performed by the service through the defined interface.
* Port – specifies an address for a binding. Defines a communication port.
* Message – describes the names and format of the messages supported by web service.
* Types – defines the data types used by the service for sending messages between the client and server.
* Binding – defines the communication protocols supported by the operations provided by the service.
* Service – specifies the address (URL) for accessing the service.

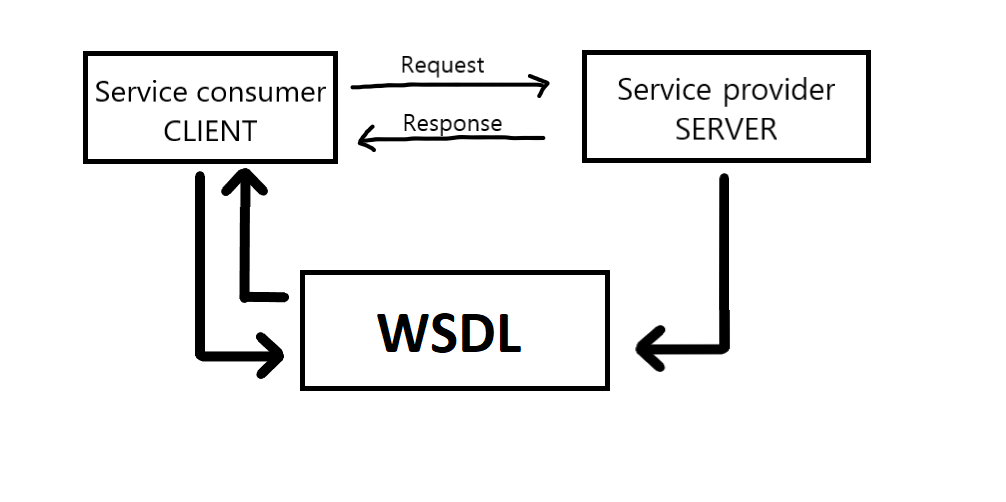
****

Figure 2 - Simple WSDL example

# UNIVERSAL DESCRIPTION DISCOVERY AND INTEGRATION - UDDI

## **What is Universal Description Discovery and Integration?**

UDDI is a standard sponsored by OASIS (Organization for the Advancement of Structured Information Standards). UDDIs are often described as the yellow pages of Web services. UDDI is a specification for creating an XML-based registry that lists information about businesses sand the Web services they offer. It provides businesses a way of listing their services and discovering services offered by other organizations.

UDDI implementations vary, UDDI often describes services using WSDL and communicates via SOAP messaging. Registering a Web service in a UDDI registry is an optional step, and UDDI registries can be public or private.

To search for a Web service, a developer can query a UDDI registry to obtain the WSDL for the service he/she wishes to utilize. Developers can also design their Web services clients to receive automatic updates about any changes to a service from the UDDI registry.

UDDI is a registry standard for Web services providers to publish their Web services. It may be used by a Web services consumer to discover (search) Web services developed by Web services providers. The role of UDDI as a service registry along with its relationships with service consumers and providers in the context of service-oriented architecture is depicted in picture below [Figure 3] [9].

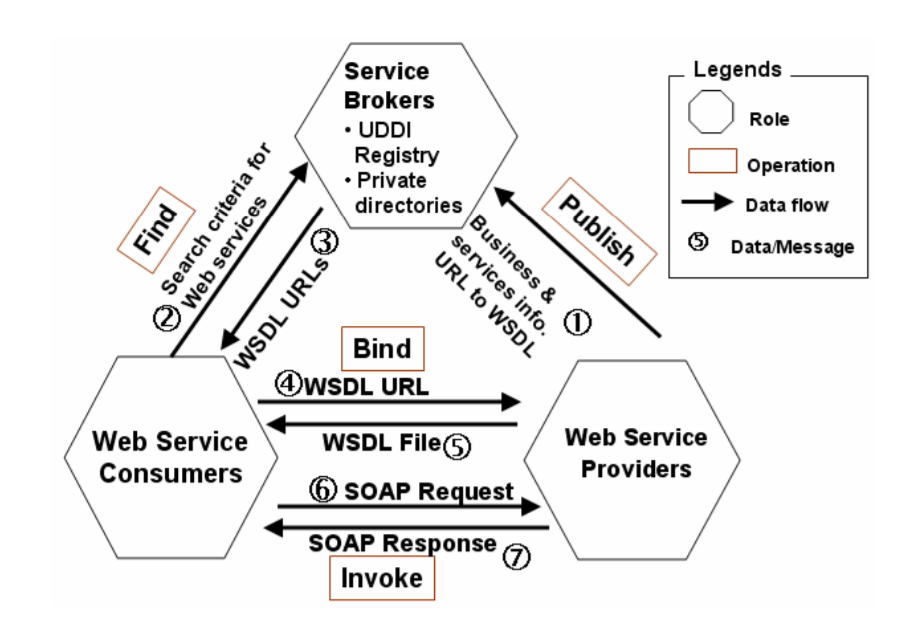


Figure 3 - Web Services-Based Architecture

## **Advantages of UDDI towards E-Commerce**

UDDI can store company information, services provided by a company, and the specific technical information for binding with a specific service. The technical binding information for using a Web service will be the URL reference to the WSDL file of the Web service.

# CONCLUSIONS

In conclusion web services play a vital role in e-commerce. Content management system is one of the most important components of e-commerce, it helps manage and modify content on the e-commerce website. Web services help building an ecommerce website, they simplify application design, provide users reusability and flexibility. They provide businesses with a standard way to communicate.

Web services replace complex, proprietary programming interfaces with Extensive Markup Language (XML) documents and standard application protocols. XML is used to tag data, and SOAP or REST is used to transfer it. Web Services Description Language (WSDL) describes the services needed and Universal Description Discovery and Integration (UDDI) tells applications what services are available.

# LITERATURE

1. Shopify. 2020. Ecommerce Definition - What Is Ecommerce. [online] Available at: <<https://www.shopify.com/encyclopedia/what-is-ecommerce>> [Accessed 6 April 2020].

2. 10. Fringe Media. 2019. *Content Management System*. [online] Available at: <<https://fringemedia.ca/top-10-content-management-systems/>> [Accessed 6 April 2020].

3. Doofinder Site Search. 2018. *What is E-Commerce and Content Management System (CMS)?* [online] Available at: <<https://www.doofinder.com/en/blog/cms-ecommerce>> [Accessed 6 April 2020].

4. Docs.aws.amazon.com. 2013*. Ecommerce and Web Services definitions* [online] Available at: <[https://docs.aws.amazon.com/AWSECommerceService/latest/DG/prod-adv-api-dg.pdf](https://docs.aws.amazon.com/AWSECommerceService/latest/DG/prod-adv-api-dg.pdf%20) > [Accessed 6 April 2020].

5. Researchgate. 2005. *Web services: E-commerce partner integration.* [online] Available at: <<https://www.researchgate.net/publication/3426717_Web_services_E-commerce_partner_integration>> [Accessed 6 April 2020].

6. WooCommerce. 2016. *Sitemaps For Ecommerce Explained.* [online] Available at: <<https://woocommerce.com/posts/ecommerce-sitemap-explained/>> [Accessed 6 April 2020].

7. Irt.org. 2000. *Perspectives Of XML In E-Commerce.* [online] Available at: <<https://www.irt.org/articles/js215/>> [Accessed 6 April 2020].

8. Clarity-ventures.com. 2020. *SOAP Ecommerce Integration And Development*. [online] Available at: <<https://www.clarity-ventures.com/resources/business-ecommerce/integrate-soap-into-your-ecommerce-business>> [Accessed 7 April 2020].

9. Pdfs.semanticscholar.org. 2003. *The applications and impacts of web services.* [online] Available at: <<https://pdfs.semanticscholar.org/698e/fa2bfd0afa2cd5d70de80c355fd720f9ce07.pdf>> [Accessed 7 April 2020].

10. Supportline.microfocus.com. *Web Services Concepts. WSDL*. [online] Available at: <<https://supportline.microfocus.com/documentation/books/nx31sp1/wsconc.htm>> [Accessed 7 April 2020].

11. W3.org. 2009. *Web Services*. [online] Available at: <<https://www.w3.org/DesignIssues/WebServices.html>> [Accessed 7 April 2020].

12. Cse.iitd.ernet.in. 2018. *Web Services – Amazon Ecommerce Service, Comparison With Other Web Services*. [online] Available at: <<http://www.cse.iitd.ernet.in/~bhalla/WEBSERVICES2.pdf>> [Accessed 7 April 2020].

13. Ibm.com. 2000. *Web Services: Taking E-Business To The Next Level*. [online] Available at: <<https://www.ibm.com/developerworks/cn/websphere/download/pdf/e-businessj.pdf>> [Accessed 7 April 2020].

14. Flylib.com. 2002. *SOAP, Web Services, And E-Commerce | XML, Web Services, And The Data Revolution*. [online] Available at: <<https://flylib.com/books/en/2.170.1.60/1/>> [Accessed 7 April 2020].

15. Namroud, P., 2019. *EDI, XML Or Web Services?.* [online] Edi2xml.com. Available at: <<https://www.edi2xml.com/blog/edi-xml-or-web-services-whats-best-for-ecommerce-integration/>> [Accessed 7 April 2020].

16. Web.csulb.edu. [online] Available at: <<https://web.csulb.edu/~pnguyen/cecs475/pdf/web%20services.pdf>> [Accessed 7 April 2020].

# SOAP & WSDL WORDPRESS PRACTICE

This short section is dedicated to show how SOAP and WSDL web services work. Basically, in short, we are using WP SOAP API plugin to expose WordPress core methods via SOAP web services. With the exposed WSDL which describes SOAP API methods from our page, we can see and access the information. I will use application called Postman to work with API calls.

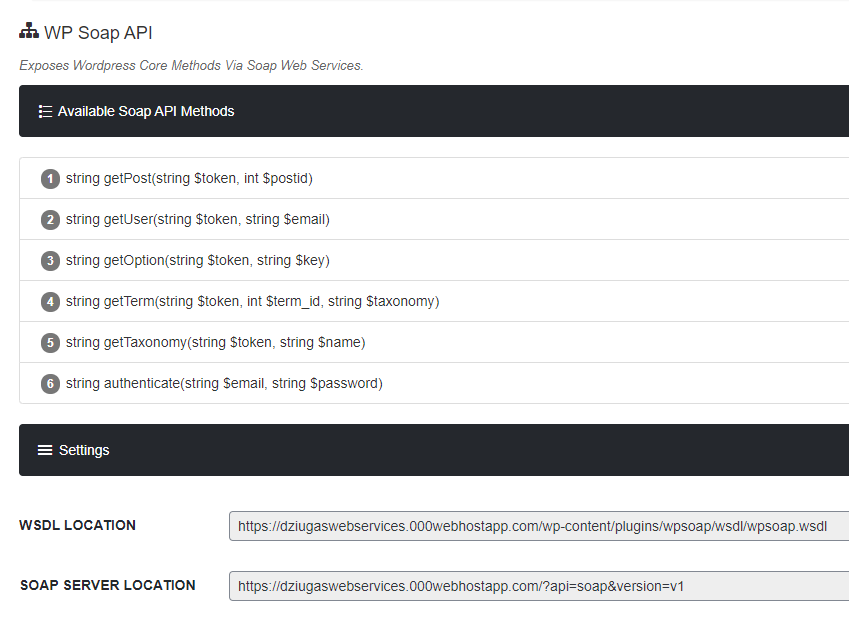


Figure 4 - SOAP API

## **Authentication token**

So to work with API methods, we will need an authentication token, we can access it via SOAP server, but firstly I want to show, that WSDL file contains all the SOAP API methods.

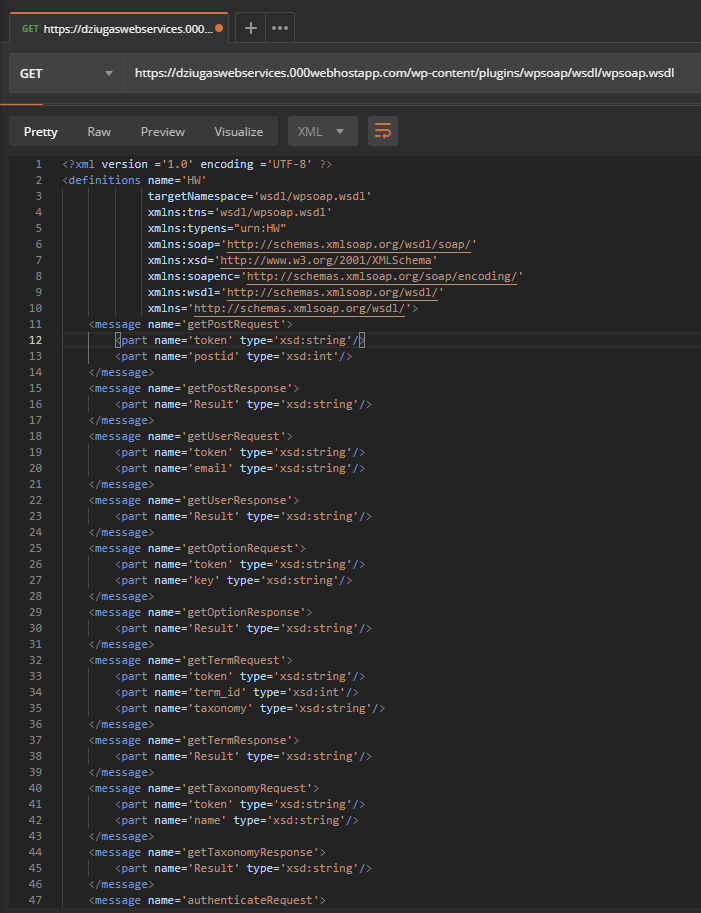


Figure 5 - WSDL message 1

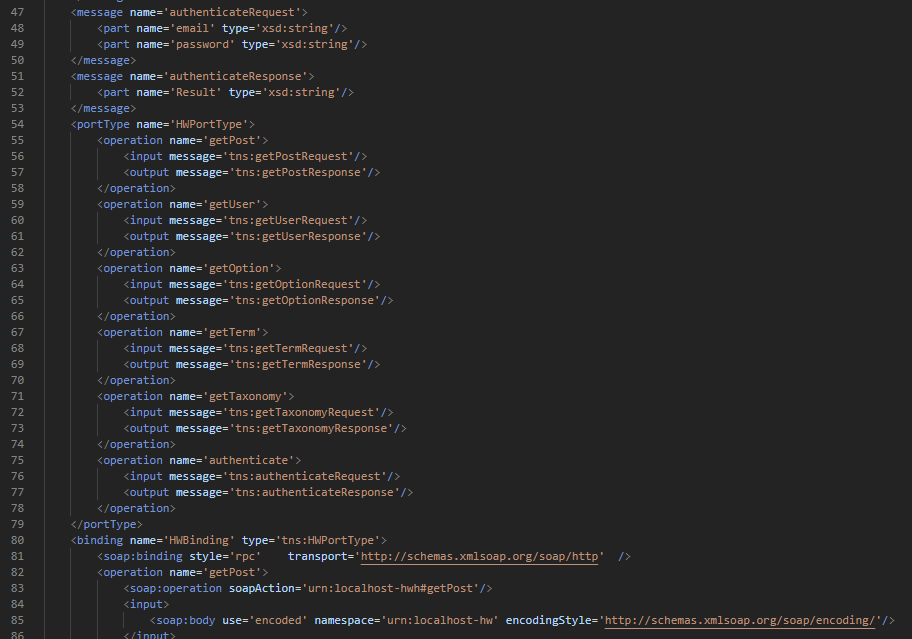


Figure 6 - WSDL message 2

Above we can see our exposed SOAP API methods from our WP page. And at the bottom of the WSDL file we should see SOAP server address.

Unfortunately, it does not show in my request, so luckily we have our server location shown in our plugin page.

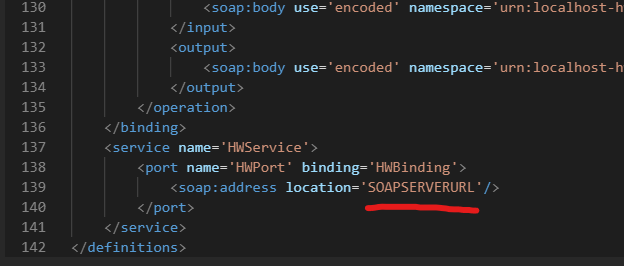


Figure 7 - SOAP Server

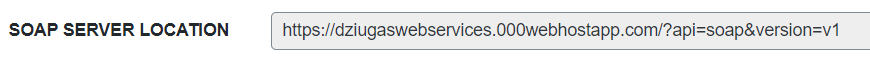


Figure 8 - SOAP SERVER LOCATION

Now once we have our SOAP server location, we can now make a POST request to it, to get our authentication token so we could access other SOAP API calls later on.

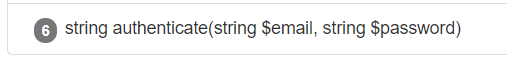


Figure 9 - authentication form

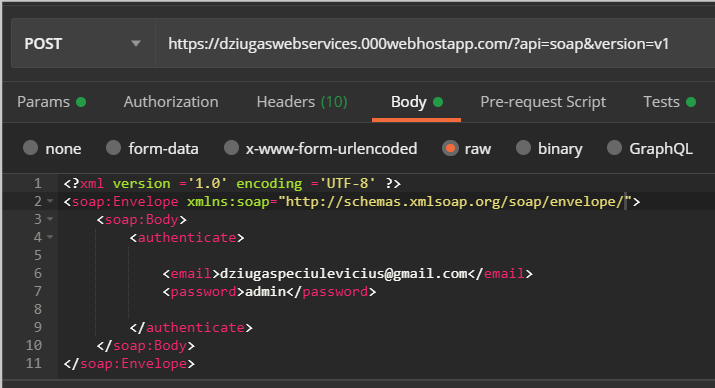


Figure 10 - authentication call

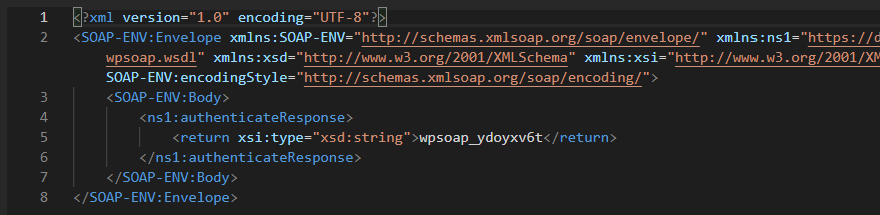


Figure 11 - authentication response

And there we go, at the bottom screenshot, we can see our authentication response with our token which is - **wsoap\_ydoyxv6t**

## **Getting a post**



Figure 12 - get post form

Since we retrieved our token we can now make a POST request to a SOAP server to get one of our existing posts. Posts can be found here, so we can see which id we would want to select before making a call: <https://dziugaswebservices.000webhostapp.com/wp-json/wp/v2/posts>

I used JSON formatter to parse the file and we could read it.

I used a different token, because the old one expired. After making a POST request we get a selected post as a response in an XML format.

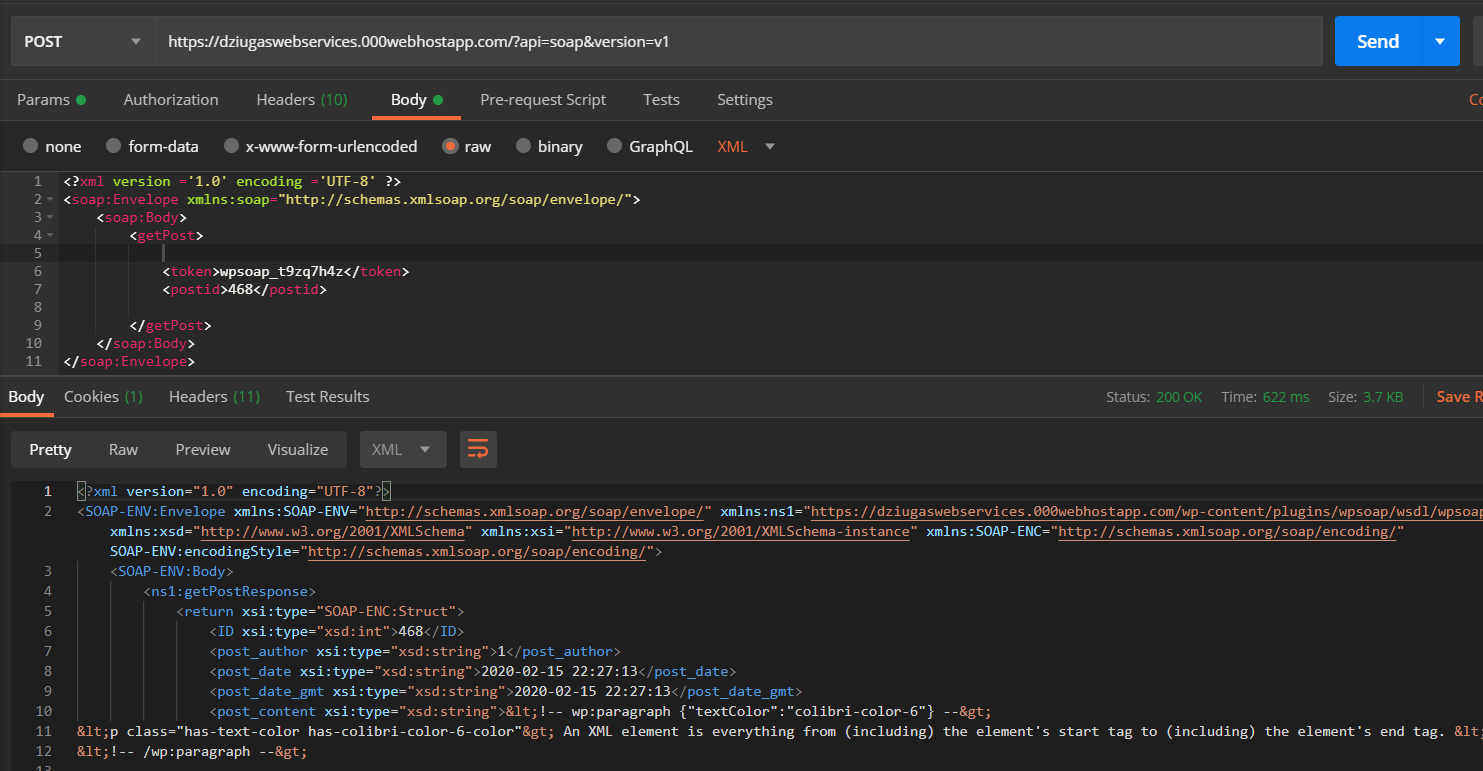


Figure 13 - get post call

Almost at the bottom of the response we can find a post link, if we click it, it will deliver us straight to the WP post.

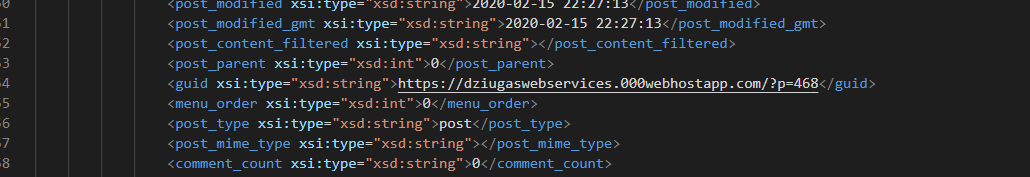


Figure 14 - WP post link

## **Getting a user**



Figure 15 - get user form

It’s a similar thing with user like we did with a post. We want to use our token and a user email to get information. To see what users are available we can see here: <https://dziugaswebservices.000webhostapp.com/wp-json/wp/v2/users>

This POST request will return information about our user that was requested.

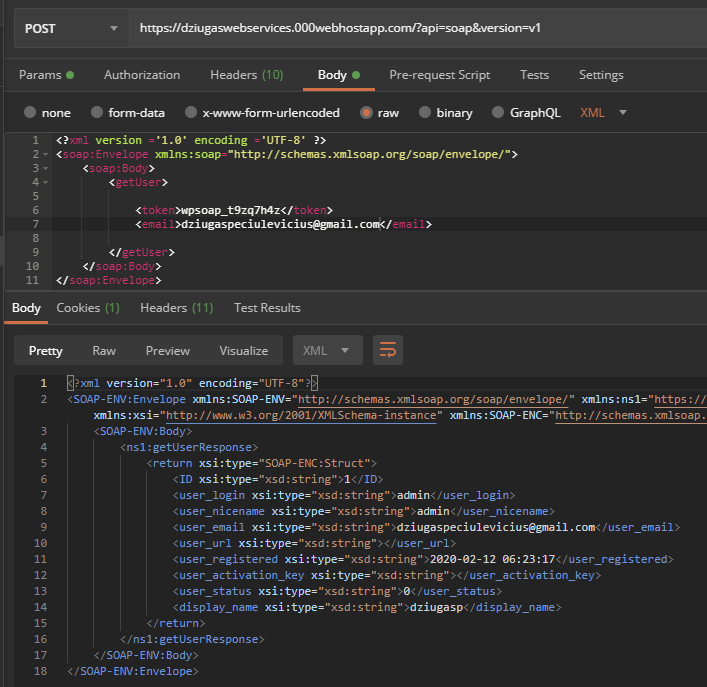


Figure 16 - get user call and response

## **Getting an option**



Figure 17 - get option form

For getOption, we already have a token, now all we need to find is what kind of keys we could use to make a request. More about the actual keys here: <https://developer.wordpress.org/reference/functions/get_option/>

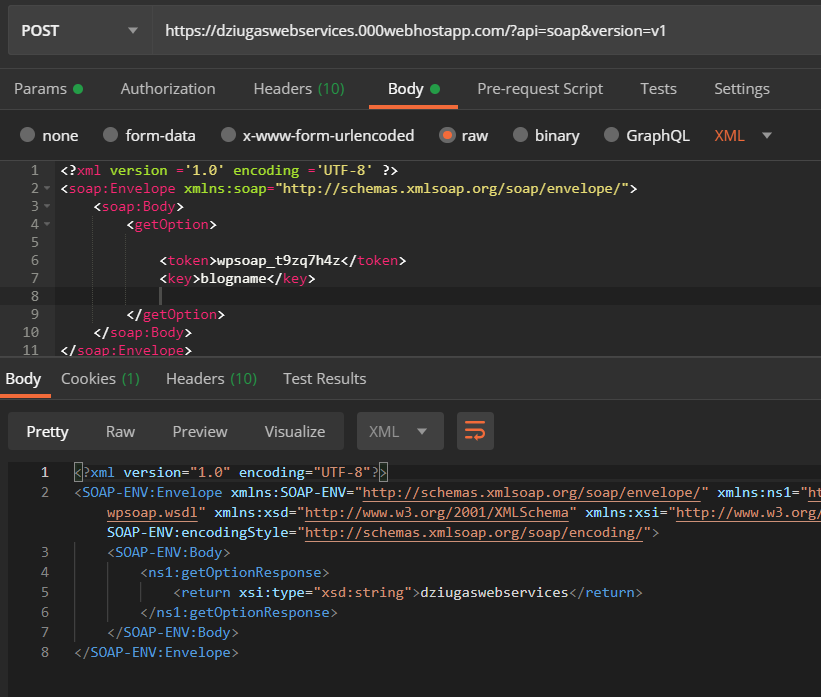


Figure - get option call and response

## **Getting a taxonomy**



Figure 19 - get taxonomy form

Taxonomies in WordPress is a classification or grouping of data (**posts and custom types together**).

There are two types of taxonomies – categories and tags. These are default taxonomies built in by WordPress. Custom taxonomies can be either category custom taxonomy (if they have a parent), and tag custom taxonomy (if they don’t have a parent).

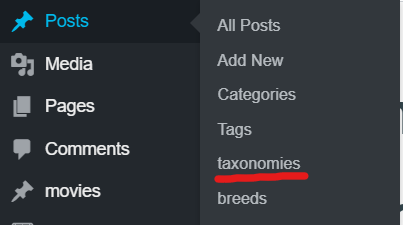


Figure 20 - created taxonomy

I used Custom Post Type UI plugin (Admin panel for creating custom post types and custom taxonomies in WordPress), and I created a taxonomy called ‘taxonomies’. (I know, so original).

And now since I have a taxonomy, we can go back to our Postman application

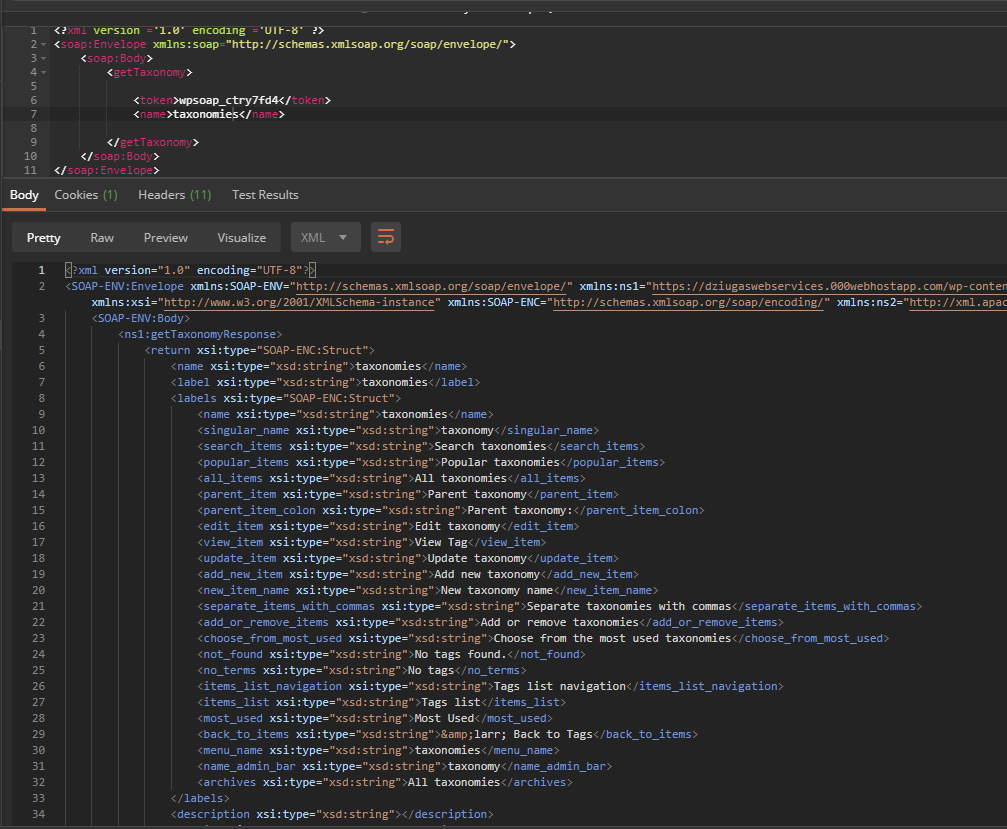


Figure 21 - get taxonomy call and response

And here we go, we get our response to a getTaxonomy request.

## **Getting a term**



Figure 22 - get term form

In WordPress, terms refer to the items in a taxonomy (which I have one of them in the list and it’s called – taxonomy1). For example, a website has categories like books, politics, and blogging in it. While category itself is a taxonomy the items inside it are called terms. To get a term request we will need three things, an authentication token, term\_id (which is an id of the taxonomy post) and a taxonomy name.

So I already have an authentication token and a taxonomy name which is ”taxonomies”. Now all I need to find is a termID. All I need to do is just go to my custom posts and find the post I need. And then we can easily find an ID we need:



Figure 23 - getting term\_id

And now I can just make a request for getting a term:

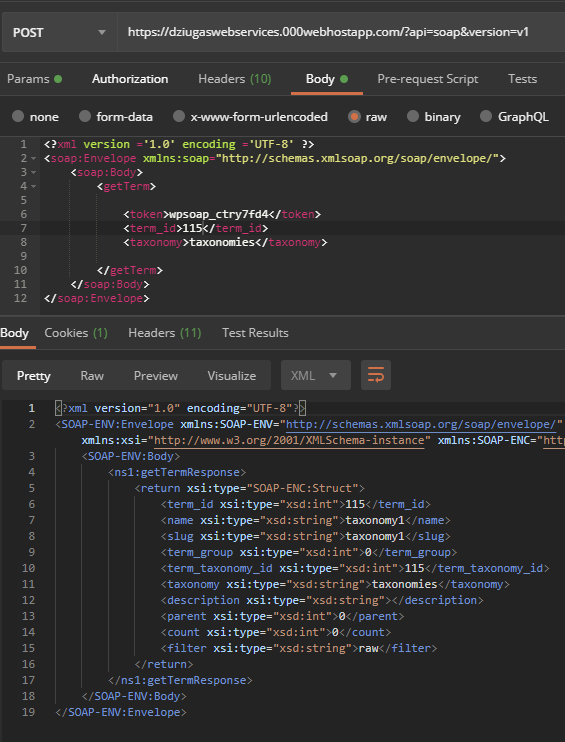


Figure 24 - getTerm call and response