Web Application Development – CA 1 – André Pont

# Develop a web application with XML, XSL, HTML,CSS & JavaScript

Content

[Develop a web application with XML, XSL, HTML,CSS & JavaScript 1](#_Toc181649342)

[Creating my first web application. 1](#_Toc181649343)

[Weekly progress 2](#_Toc181649344)

[The first week of work (14 - 19 Oct) 2](#_Toc181649345)

[The second week (20 – 26 Oct) 3](#_Toc181649346)

[The third week (27 Oct – 2 Nov) 4](#_Toc181649347)

[The final week (3 – 9 Nov) 5](#_Toc181649348)

[Conclusion 6](#_Toc181649349)

[Reference list 7](#_Toc181649350)

# Creating my first web application.

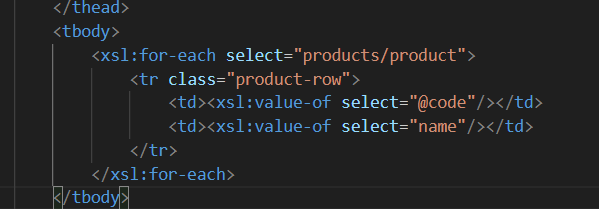
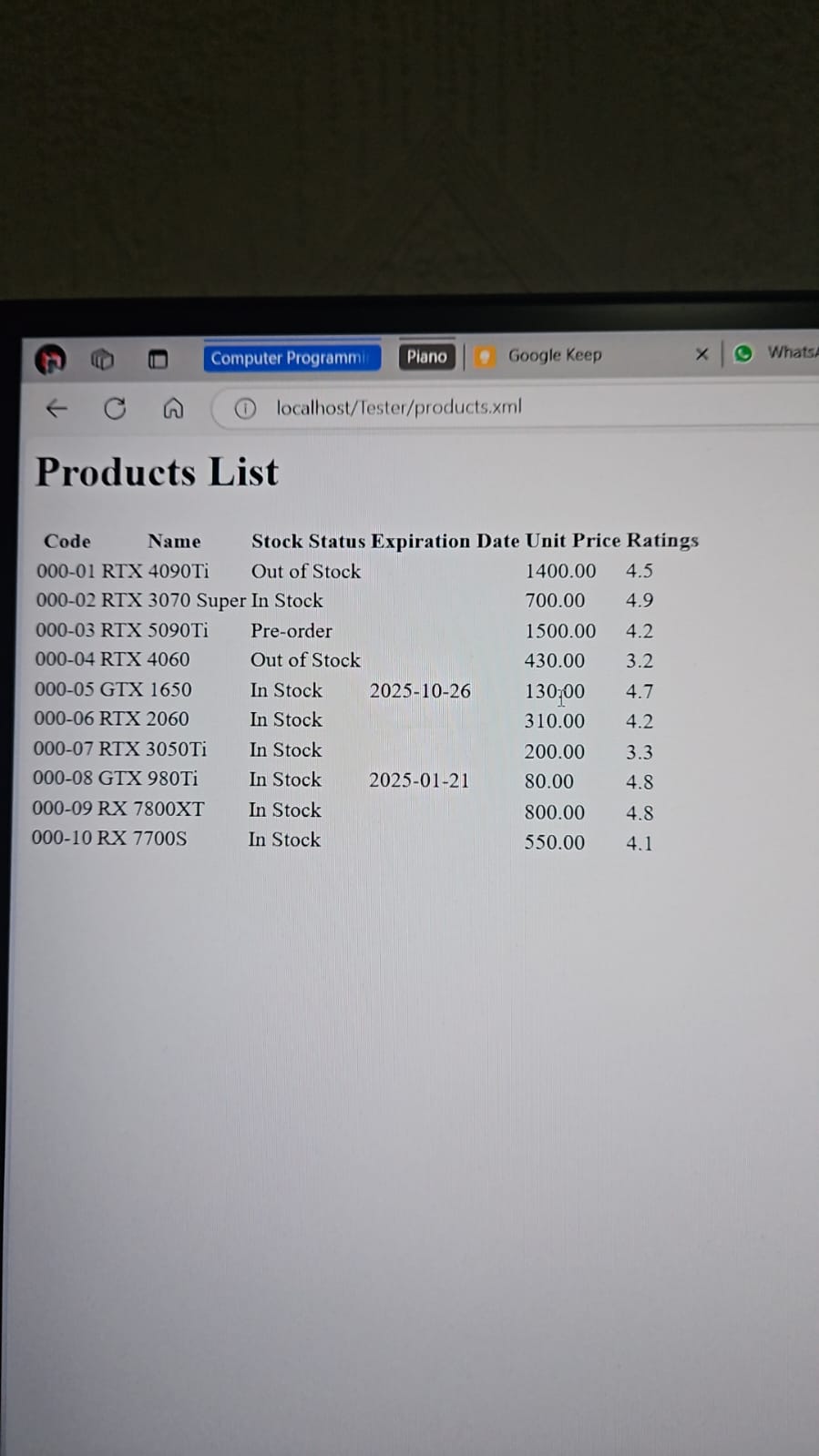
The chances that the reader, all his/her friends, and the vast majority of his family has used a form of web application in the last hour is more likely to happen than not. Since, according to datareportal (*Internet use in 2024*. [online] DataReportal*)*, the average internet user spends approximately 6 hours on the internet daily. Knowing this, it is safe to say that web applications and web pages are one of the backbones of modern civilization, and being able to access a service without necessarily downloading it is a commodity that few would be willing to give up.

While the great majority of the population uses some sort of web application during the day, it is also the majority that does not know how they work, in this assessment I will be trying to answer that question with a simple website that is able to retrieve information from one of the most basic techniques on the internet: XML.

# A screen shot of a computer screen Description automatically generatedWeekly progress

The first week of work (14 - 19 Oct) I decided to simply create the layout of the page, I made a Graphics Card store site that welcomes you with a main page and then directs you to the products page where you can find the product lookup tool, I also made a “about us” page to make it feel more like a real website and applied some basic styling so it was easier to work with.

Once the layout of the web page was ready, I started to add information to the xml file, I added graphics cards of different generations and separated by manufacturers (AMD and NVIDIA), naming the cards was an easy task as I used real-life models for reference. On the other hand, for the product code, I had to think of a coding system that made sense according to each card, at the end I decided to implement it in a way that each index would reference a property of the card, for example 1xx-xx for NVIDIA 2xx-xx for AMD, or NVIDIA 40 series: 14x-xx and also Ti or XT variants get a 1 as the third digit (141-90 for an NVIDIA 4090 Ti). It was a slow start, but the progress made this week would serve as the outline for the next few weeks.

The second week (20 – 26 Oct): This week was mainly focused on developing the XSL page, and while I knew how to use the basic html functionalities, I had trouble merging the xml with the html itself, what I wanted to do, was to have a list of every product code and name on the left of the screen and on the right side to have the tool to display the full information of the product when searched, after refreshing my knowledge with my notes from the XSL lecture, I managed to use a for loop to display each product on a table, while maintaining its format.

While the code was working, and I was able to see my products without an issue, I still needed to validate my code through an XML Schema, so I started developing my xsd file, and while developing the code, I noticed that it was getting too complex and hard to read because I was doing it in a single complextype instance, which was probably simpler to develop, but having a code being easy to read gives us an enormous advantage. So I changed it to be able to individually address every element of the XML.

In the picture in the right it is possible to see how restrictions are still possible and there are not any compromises on how well the data is validated.

A screen shot of a computer

Description automatically generated A screenshot of a computer program

Description automatically generated

A screenshot of a computer

Description automatically generatedThe third week (27 Oct – 2 Nov): On the third week, I focused on developing the JavaScript for the functionality of search tool, which was by far the most complicated piece of code to develop of the entire assignment, as I had to do a lot of research on my own, and almost every web page that I managed to find had a different approach to doing this and had no concrete explanation on how to do it.

After looking at a number of different resources, I finally found w3School, which had clear explanations on how to parse an XML file into a XML DOM object, which then allowed me to manipulate the data on my JavaScript in order to be able to locate the desired product on the search tool. This made the making of the script way easier and just by looking through more examples and documentation I was finally able to finish it.

A screen shot of a computer code

Description automatically generatedIn my approach, the code is run whenever the search button is pressed, it then retrieves the inputed product code from a text box and stores it in a variable. After that, it fetches the raw xml file to extract the text and run it through the parser with the parseFromString function.

A computer screen shot of a program

Description automatically generatedThe path is then stored in a variable with the inputed code from the user and it is later compared to every product code in the xml file with the evaluate function, if there is a match, it retrieves the information of the specified product and changes the innerHTML of the display section of the products page.

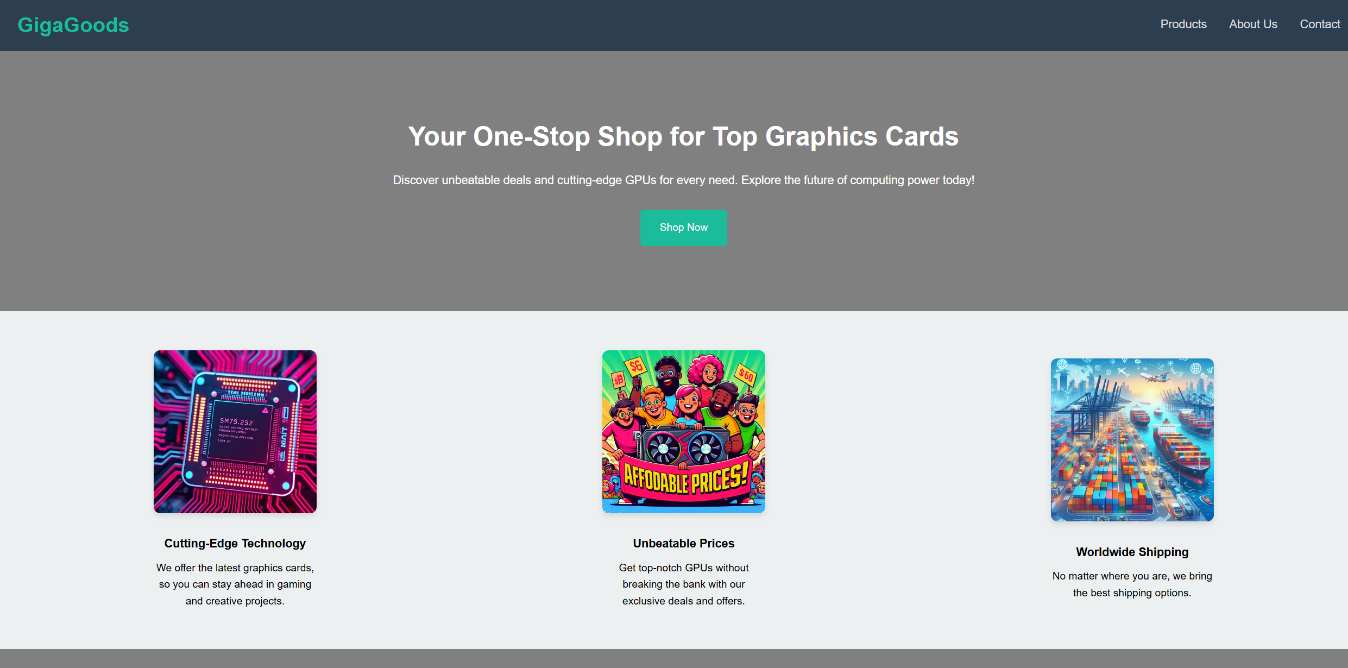
Overall, the key aspect of this specific search tool was to understand the xPath functionality. I had knowledge of how xpath worked before starting the script, but the challenge was to know how to use that knowledge and implement it into code.

A screen shot of a computer screen

Description automatically generatedThe final week (3 – 9 Nov) : This week, I solely focused on the appearance of the web page, putting to the test my CSS skills, I decided to give it a more professional look and have the entire site have a simple colour pallet and a focus on the functionality, making it easy to read and understand all the content.

I used as reference some open-source pages that w3School had as templates. This helped tremendously as I had not used CSS for a long time and looking at professionally developed pages served as a fantastic way to get back into it.

The result was very eye-catching, and the CSS was properly ordered into related sections and commented to make it easier to work with.



While my page was finished, and the JavaScript file was working, I decided to ask the lecturer if my implementation of the JS was correct, and he illustrated to me that while it was functional and well made, I had to do it with what we had covered in class.

After learning about this, I then decided to redo the JavaScript, It was simpler, but still just as functional, It retrieved all of the information from the HTML document that was created with the XSL instead of from the XML directly, but since I did not want to show all of the information in my catalogue table, I simply hid every column I did not want the user to see and then retrieved the information as usual

# Conclusion

In this project, I understood the complexity of web development a little further, specially when it comes to integrating multiple technologies to build an application that is cohesive. Using XML and XSL might not be the most efficient way to manage the data anymore, but it served as a valuable lesson on the basics of how to create and manage a web application, and how product listing might work for some companies.

Every step represented its own challenge, especially JavaScript, as it reminded me that knowing how to code is just as important as knowing how to read documentation and adapting to new challenges to try to not stay behind on this competitive area known as computing. But overall, finding solutions to problems is the skill that is the most important here, and this exercise helped me refine this and deepen my understanding on all the technologies involved. Moving forward, I am better equipped to tackle similar projects with increased confidence and efficiency.

## Reference list

Kemp, S. (2024). *Internet use in 2024*. [online] DataReportal – Global Digital Insights. Available at: <https://datareportal.com/reports/digital-2024-deep-dive-the-state-of-internet-adoption>.

W3Schools (2020). W3.CSS Templates. [online] W3schools.com. Available at: https://www.w3schools.com/w3css/w3css\_templates.asp.

www.w3schools.com. (n.d.). XML Parser. [online] Available at: https://www.w3schools.com/xml/xml\_parser.asp.

www.w3schools.com. (n.d.). XPath Examples. [online] Available at: <https://www.w3schools.com/xml/xpath_examples.asp>.

Las mejores fotos de stock gratis, imágenes libres de regalías y vídeos compartidos por creadores. (n.d.). pexels.com. Available at: https://www.pexels.com/es-es/.

‌