# **From Web to Desktop**

A Focal Shift in Development

Henry Larson

Computer Science Department

Colorado State University

Fort Collins

hhl2048@colostate.edu

###

## **ABSTRACT**

The first programs (as we would think of them today) were all stored locally on a computer as an executable. Then the web came around and enabled people to display data online to anybody who had an internet connection. Finally, these two ideas merged. Why store something on your machine when you can just have it online for anyone to access. In today’s world, there are both web and desktop applications for many programs. Discord, Apple, Alibaba, and Spotify are a few companies that have both web-based and desktop-based apps. However, the issue with web applications has always been speed. The same app downloaded on a computer will respond and function more quickly than a web app will. Although there have been large strides in data transmission speeds and increased focus on web-based development, the ability to interconnect computers to the web has become so fast that web apps are being accessed less each day. This short paper will explore how and why this shift is occurring, the boons it may bring to our society, and what it means for the internet as a whole.

## **1 THE SHIFT**

Computers and software have come a long way since the first days of digital computing (Bychkov). Programs were always run as an application on a machine until the World Wide Web came along. Then we had the ability to display content on a web browser as opposed to just on a graphical OS. These changes in development altered the focus in development from just executable applications to web-based applications as well. Having an application running on a website allows for a computer to do less heavy lifting since there is a server that has most of the resources you need. Especially in the modern age where apps can have full-fledged online programs as opposed to machine-based programs. Software like PayPal, Alibaba, Netflix, and Amazon can (and are mostly) run as websites as opposed to desktop applications. While other applications with both websites and desktop apps (applications) such as Discord, MS Office, and Spotify are run as downloaded executables. We are in an interesting interim zone where some apps are mostly run locally on machines while others are mostly run as websites. But where does this preference stem from?

## **1.1 DEVELOPMENT STYLE**

This preference for using a web app or a desktop app most probably stems from how that application was programmed. For example, Netflix was developed using React Native which is an open-source JavaScript framework, among many other programming languages (Poudel). This means that for Netflix to have a desktop application (which they do) they have to rebuild their site as a desktop app. Converting a massive codebase takes a lot of time and work from a new team of desktop-focused developers. Additionally, this means that the web-based developers and desktop-based developers have to communicate with each other on what features to add or remove and how to implement them. Long story short, more people probably use the Netflix website because there may have been less effort put into the desktop application. Discord, on the other hand, is mainly programmed using Python, Rust, and C++. Which allows them to more easily convert their web-based code to desktop-based code. There is something else to think about here as well: what is the purpose of the application and how do users use it?

## **1.2 PURPOSE & THE USER**

Discord is a service based on text, audio, and video chatting. So it makes sense that most people would rather use the desktop application. Users want fast and reliable voice chat which is something a website can’t as reliably provide as a desktop app can. On the other hand, Amazon is a service built around eCommerce and its website clearly reflects this. Because of the service amazon is designed for there isn’t a huge difficulty in making a desktop (or mobile device) application. Since the website is more or less a series of menus and some backend it would be considerably more simple to make a desktop counterpart for the website. It should also be stated that a desktop application can be used anywhere (if it doesn’t require a connection to the internet). For example, MS Word as a desktop application can be run without a connection to the internet. So it is more convenient to have that app downloaded as opposed to relying on having an internet connection to access the website.

## **1.3 THE FUTURE**

I mentioned earlier that there is a sort of grey zone between applications on the web versus applications on a desktop. This brings me to a more opinion-oriented portion of the essay where I will discuss why I think this is and where we are headed. In the past, an app on a desktop wasn’t necessarily tied to the internet. But now, the internet is more closely interwoven with users' devices. As the world becomes more connected via the internet more devices require it. So, I think this grey zone comes from the transition to a more interconnected globe. As more and more smart devices emerge so too must the number of signals increase to support those devices. If the internet becomes more widely available then we might start to see more people using web apps as opposed to desktop ones. Additionally, as data transfer rates get faster with the onset of 5G the internet can be more rapidly accessed and used. We are seeing the beginning of this shift with many apps using a centralized cloud network. Google and MS Office, for example, both have a central storage location in the cloud. To access this you need to have an internet connection.

## **SUMMARY & CONCLUSIONS**

So, where does this bring us? Computers went from being able to only have programs stored on their disk to be able to use a web format as well. In the present day, we see many applications that have both a web app and a desktop app. Plus, we are seeing many desktop apps use the internet to a greater extent. As websites become more sophisticated and highly developed (Jazayeri) we might begin to see less focus on desktop applications and a shift to more website applications. Every day, our world comes more closely connected with new technologies and data transmission methods. One can speculate as to the direction development will move in the coming years. But, I think that there will be a greater focus on tying the web to a user's device and we will see that reflected in the applications released.

## **REFERENCES**

1. Bychkov, D., 2022. Desktop vs. Web Applications: A Deeper Look and Comparison. Segue Technologies.
2. Chen, B., Hsu, H. and Huang, Y., 2016. Bringing Desktop Applications to the Web. IT Professional, 18(1), pp.34-40.
3. Jazayeri, M., 2007. Some Trends in Web Application Development. Ieeexplore.ieee.org.
4. Torchiano, M., Ricca, F. and Marchetto, A., 2010. Are web applications more defect-prone than desktop applications?. International Journal on Software Tools for Technology Transfer, 13(2), pp.151-166.
5. Weiss, A., 2005. WebOS. netWorker, 9(4), pp.18-26.
6. Poudel, D., 2022. Technologies and programming language used by Netflix. Ourtechroom.com.