

Editorial

Why physicians might want to learn computer programming

Pieter L. Kubben

Department of Neurosurgery, Maastricht University Medical Center, The Netherlands

E-mail: *Pieter L. Kubben - pieter@kubben.nl

*Corresponding author

Received: 19 February 13 Accepted: 20 February 13 Published: 22 March 13

This article may be cited as:Kubben PL. Why physicians might want to learn computer programming. *Surg Neurol Int* 2013;4:30.Available FREE in open access from: <http://www.surgicalneurologyint.com/text.asp?2013/4/1/30/109461>

Copyright: © 2013 Kubben PL. This is an open-access article distributed under the terms of the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original author and source are credited.

How long does it take to become a physician? At least 6-7 years in most countries. Then, after finishing medical education, it takes another 3-6 years to do a medical specialization. Now, as a comparison: How long does it take to become a computer programmer? The answer is 3-4 years for a professional programmer. An enthusiastic amateur can grasp the basics in less time.

This short introduction is not meant to bash computer programmers. Good programming is a skill in itself and requires practice, just like any other skill. But for obvious reasons it makes sense to state that the learning path toward being a clinically experienced physician is longer than a path toward being a computer programmer. Second, it is hard to have some kind of “do it yourself” medical training because the associated responsibilities do not allow that. It is not hard at all to buy/download a book on computer programming, or watch an online video tutorial, and get some hands-on experience. Therefore, in my opinion, it does make sense to teach a physician how to do computer programming. Let me tell you my story...

When I started medical education, I liked computers. That was 1996. I did not have that much computer experience, but liked to experiment a little. When I started using Internet, soon I wanted to build a webpage myself, and I bought a book on hypertext markup language (HTML) (the web’s language). Reading the text, trying some exercises, I found that this was no magic. This was doable. Then I wanted a little more interactivity, and found JavaScript. Did a course in Delphi programming, just for fun (yes, I am a little crazy). Then I wanted even more interactivity for my website, and started learning the basics of PHP (an open-source scripting language) and MySQL (an open-source and

very powerful database format). Using PHP and MySQL, I built a web-based framework for online collaboration, including an output format that was optimized for mobile devices. This was 2004, iPhones did not exist yet, neither did “apps”. But it worked, and I won some prizes. The key to the success was simple: Scratch your own itch. I had a problem remembering some things that I occasionally needed in my clinical work, which I started back in 2003 at the department of neurosurgery. Others turned out to have that problem too. Then in 2008, I bought my first iPhone, running iOS (then still iPhone OS) version 1.2. That is old. You could not even move icons on your springboard. But more important: Only with version 2.0 something called the App Store was introduced. That gave me a choice: Either I could create a list with “things to remember” for personal use, or I could learn iPhone programming and make an “app” for this. I chose option number two. The app is called NeuroMind, and started with mainly classification systems that I considered useful to carry in my pocket. Or to have other people carry them in their pocket. The app was launched early 2010, and immediately got attention on social networks (mainly Twitter and some web logs). Downloads went up, and soon this was – and remained – the most downloaded neurosurgical app in the world. Later in 2010 I released a version for iPad and

Access this article online

Quick Response Code:



Website:

www.surgicalneurologyint.com

DOI:

10.4103/2152-7806.109461

Android, which in turn boosted downloads even more. The fact that I kept the app completely free of charge is also helpful, of course. In 2011, I was invited to deliver a Technology, Entertainment and Design (TED) talk on my vision regarding computerized clinical decision support systems and mobile health, in 2012, NeuroMind was upgraded to version 2.0 that actually includes interactive decision support (and now more than 100 neurosurgical classification systems, many of which I still cannot remember), and in Summer 2012, I was invited by Intel to port NeuroMind to Windows 8, which has been done. The app now gets serious attention, also from magazines, and I was fortunate to be named one of the ten Internet-smartest doctors in the world on a larger medical weblog.

So, why learn computer programming? First, because it is fun if you like to combine creativity, puzzling, and having a useful result in the end. Second, because it can help to boost your career: The results can be a great business card that is spread automatically. You prove your ability to translate ideas into results. There are many other ways to do that, but this is one of them. And expecting an even higher dependency on information technology in the future, which might be worth the effort, because we do not want technicians to decide how we do our job.

But somebody has to talk to them. Preferably somebody who (partially) understands their language. They would neither understand yours, nor will they understand how things go in daily practice. Three, because if you scratch your own itch, it is likely that other people may have that itch too. They will be interested in what you have, whether that is a weblog with information (no programming knowledge required), a simple webpage, or an app. Feel free to charge money if you think your service is worth that, but keep in mind that less people will use your product then. Work out a (business) model that works for you, your service, and your career. And realize that if you needed to hire a programmer, the whole project would not even get off the ground. As physicians, we master the content part. If we can get some skills on how to distribute this content, we have a winning combination. And with modern webbased technology, the latter is neither difficult nor expensive.

There is one thing that you should avoid, and that is doing nothing now and complaining later “I had that idea, and if I had done that, I would be successful now”. Such “ifs” do not count, they just demonstrate a lack of character.

Happy programming!