From: James Pirruccello [e-mail redacted]
Sent: Sunday, September 26, 2010 3:35 AM
To: Bilski_Guidance
Cc: [e-mail redacted]
Subject: On software patents (a medical student's perspective)

Dear USPTO,

I am first and foremost a medical student and a future physician. But nearly every night, I steal an hour or two from my medical studies to write software. I believe that, in the end, learning to program will make me a better physician. But I also need motivation to learn, and my motivation comes from writing software that is used by hundreds of thousands of people each year.

Software is also an escape for me; medicine is uncertain, but software is not. I cannot always know whether my diagnosis is correct, or how a particular patient will respond to a treatment. In contrast, I can know exactly how the software that I write will work, because software is mathematical. Indeed, software is nothing more or less than mathematics.

As a student of medicine and an author of software, I oppose software patents for the following reasons:

1) Software is math, which is unpatentable.

2) Software is well protected without patents. Software's source code is already protected by copyright. Software that runs "as a service" over the internet (like the software that I write) is furthermore protected as a trade secret, since only its output--not the software itself--need be distributed.

3) The existence of software patents creates the risk that I, a student writing software in my spare time, might unintentionally infringe. The low barrier to obtaining software patents (see point "B" below) makes it likely that I already, unknowingly, have done so.

4) Patents are intended to be tools that serve the public good for a very particular purpose: to promote the progress of science and useful arts. This is a pragmatic purpose which should periodically prompt the
question: does the current patent system for software promote the progress of science and the useful arts? This is an empirical question that should be investigated seriously and in scientific fashion.

If you choose to retain software patents, then please consider the
following:

A) The period of patent protection should match the period of innovation. Software innovation occurs much more rapidly than innovation in physical goods. Software patent protection should last only long enough to stimulate innovation. This is likely a period of time ranging from months to a year or two, not 20 years. To understand this timescale in the context of software development, consider that the World Wide Web was created only 20 years ago. (Fortunately, Tim Berners-Lee and his employers chose not to patent that particular invention.)

B) The barrier to software patents is far too low. See Amazon's "one click." There is nothing novel about the algorithm needed to allow one-click checkout of your online shopping cart; anyone with even minimal skills in the art, such as myself, could write such software.
This is simply good user interface design; they have reduced the barrier to purchase by eliminating all but one step. It is a clever idea, but it should be no more patentable than the clever idea of, say, installing a drive-through window at a fast food restaurant.

C) A much more aggressive approach towards identifying prior art in software should be taken. Finding whether or not prior art exists for software in the digital age is much easier than trying to identify prior art for hardware in the industrial age was.

D) The USPTO should routinely ask programmers to review software in order to discover which software patent applications should be rejected due to prior art, obviousness, etc.

Thank you for your consideration.

Sincerely,

James Pirruccello
MSIV, The Johns Hopkins University School of Medicine Sarnoff Fellow, Massachusetts General Hospital and the Broad Institute of MIT and Harvard Co-Founder, MyChances.net