

For areas where software has a chance of causing tangible harm, testing must become ever more rigorous to match the heightened peril. For more frivolous software, like video games or organizer apps, more minimal testing is acceptable and might even be a smart decision based upon business requirements. On the other hand, industries that affect the livelihoods or health of others, like medical, banking, or government, ought to have higher standards for testing and code quality. Many of these already do via regulation like FIPS and HIPPA.

These kinds of regulations and standards are consistent with the ACM code of ethics' requirement to "Recognize and take special care of systems that become integrated into the infrastructure of society" (ACM Code of Ethics 3.7). This special care requirement is reinforced by Jesus' command, "Just as you want others to do for you, do the same for them" (Luke 6:31, CSB). In this instance, it can be assumed that most people would want software that affects their health or finances to be strictly tested. From this, it can be concluded that strict software testing for such critical infrastructure systems is necessary in order to follow the aforementioned command.

With that established, it is now prudent to discuss the certification of software engineers. In this case, certification should only be legally mandated for software engineers working in industries that are already regulated by legislation like HIPPA. This already has precedent from the existing regulation, which holds companies and developers strictly accountable for any harm caused by their code. Mandating certification for these industries would merely require that programmers are fully prepared to handle the increased liability their actions present. While it is possible to certify software engineers for other industries, being overzealous with requirements could unnecessarily burden projects that are too trivial to have significant tangible risks to its users.

One easy example of a tragedy that could have been averted with proper engineering practice is the THERAC-25 incident. As a one-man project written in assembly language with little documentation, the lack of proper software testing all but guaranteed that it would have many errors (Leveson 20-21). If the programmer had passed certification and had that certification on the line, he would have known to test his code and possibly request some junior developers to assist him.

As with most things, this certification should be implemented at the state level since this sort of certification would not fall under the scope of the Federal legislature (U.S. Constitution, Article I Section 8). State level implementation will also allow each state to adapt their requirements to the specific situations of their industries. As one example, West Virginia might require stricter certification for software involving the coal industry since any failures there affect the entire state. In conclusion, software in critical industries such as banking and medical must be more strictly tested and the programmers of said software must have stricter standards and certifications.

## Works Cited

Code of Ethics, Association for Computing Machinery, [www.acm.org/code-of-ethics](http://www.acm.org/code-of-ethics).

Accessed 9 Sept. 2023.

Leveson, N. G., and C. S. Turner. "An Investigation of the Therac-25 Accidents."

Computer (Long Beach, Calif.), vol. 26, no. 7, 1993, pp. 18–41,

<https://doi.org/10.1109/MC.1993.274940>.

*Christian Standard Bible*. <https://read.cs bible.com/?book=luke&chapter=6>. Accessed 9

Sept. 2023.

The Constitution of the United States: A Transcription. National Archives, U.S. National Archives and Records Administration,

[www.archives.gov/founding-docs/constitution-transcript](http://www.archives.gov/founding-docs/constitution-transcript). Accessed 9 Sept. 2023.