**Standard Software Systems Development Questions**

David M. Vermillion

Colorado State University Global

CSC 505-1

Dr. Bingdong Li

November 18, 2023

**Standard Software Systems Development Questions**

This section of the assignment contains five questions. At least two references are expected in this document. No diagrams are required for the program in the first section of this assignment.

**Question 1**

Why does it take so long to get software finished? Software is complicated. This complication leads to many possible failure points. These failure points often extend development time. Further, from a project management perspective, if the definition of finished and expected features change during development, they will also negatively impact the time it takes to develop software. Recent research indicates that half a developer's time is spent on debugging, which indicates considerable challenges in predicting development time (Britton et al., 2020).

**Question 2**

Why are development costs so high? The answer to this one ties to the first question. If you cannot adequately plan for duration, you cannot adequately plan for cost. This creates high development costs.

**Question 3**

Why can't we find all errors before we give the software to our customers? Software is complex, involving lots of code and dependencies. This results in more points of failure than can be tested for. A good practice is to test for all common and slightly uncommon possible failure modes. Customers expect further bug fixes to address unforeseen issues. This means we should build good enough software and produce fixes as soon as possible (Rue, 2021).

**Question 4**

Why do we spend so much time and effort maintaining existing programs? New bugs are discovered after software is released. Changes to dependencies also introduce new bugs. For example, a visualization program the author of this Critical Thinking Paper developed broke with a syntax change to ggplot2 that was not yet updated in an add-on package. Simply removing the option to use a critical function broke the entire program.

**Question 5**

Why do we continue to have difficulty in measuring progress, as software is being developed and maintained? Development processes borne out of chaos cannot bring order. This means that excessive time is spent fighting metaphorical fires, and little time is spent keeping the project moving smoothly with realistically definable measurement points (Kim et al., 2023). Kanban and Agile processes aim to minimize these troubles but require consistent and persistent implementation to be successful (McKenzie et al., 2019).

Conclusion

This assignment section examines five questions about what makes software development challenging. It identifies a need for clear and consistent processes to limit most common problems.

References

Britton, T., Jeng, L., Graham, C., & Katzenellenbogen, T. (2020, November). *Reversible Debugging Software “Quantify the time and cost saved using reversible debuggers.”* https://www.researchgate.net/publication/345843594\_Reversible\_Debugging\_Software\_Quantify\_the\_time\_and\_cost\_saved\_using\_reversible\_debuggers

Kim, G., Behr, K., & Spafford, G. (2023). *The Phoenix Project: A Novel about IT, DevOps, and Helping Your Business Win*. IT Revolution Press.

McKenzie, T., Trujillo, M. M., & Hoermann, S. (2019). Software Engineering Practices and Methods in the Game Development Industry. *Extended Abstracts of the Annual Symposium on Computer-Human Interaction in Play Companion Extended Abstracts*, 181–193. https://doi.org/10.1145/3341215.3354647

Rue, B. (2021, May 21). *Council Post: Why Error-Free Software Isn’t The Goal*. Forbes. https://www.forbes.com/sites/forbestechcouncil/2021/05/21/why-error-free-software-isnt-the-goal/