The Development Phase Challenges

Name of Student

Institution of Affiliation

Development Phase Challenges in Computer science.

Research shows that the idea of automated machines like self-driving cars, self-lacing shoes and smartwatches would be a big sell on the market. Computer science provides a platform for developing algorithms that run such machines. However, there are challenges that programmers face during the face of development. In this context, the paper discusses these challenges, strategies for their resolution and obstacles that may arise during the process of resolution.

The most common problem encountered during the development phase is keeping up with technology. According to Blake (2017), keeping up with technology requires more funding as the programmer would need new system specification to make an improved version of the solution. Adams & McIntosh (2016) further say that a programmer should never stop learning to be able to keep up with the dynamic technology. The learning process is somewhat cumbersome and takes much of the programmer's time. As a result, all the methods to enable the developers to keep up with the technology include increasing the project budget and allocation of more time.

The other challenge is time estimation. A programmer is expected to meet the deadline estimated for the project. According to Klemp et al. (2011), early meeting of deadlines can be a basis for placing higher price quotes for project schedules. Plans may include the debugging of codes or the development of specific classes within codes. As a novice programmer, determined to present quality work, requesting for deadline extension might be inevitable. However, this may not be the appropriate call as the teammates might complete the project before the novice programmer hence dragging them behind. Consequently, this may cause the loss of large amounts of money as well as trust.

Luckily, some strategies can be employed to solve the problems mentioned above. Keeping up with the latest trends in coding is one of them. It is hard to learn revolutionary concepts when there are deadlines one is required to meet (Lye & Koh, 2014). However, learning new codes enables the programmer to create innovative products in the future. It is therefore essential to spare some time to develop skills and knowledge.

Moreover, the best way to solve the problem on time estimation is to break down the project into smaller tasks that are manageable (Berry & Tucker, 2016). The next step is to assign each task some time limit without being biased, that is, assigning some tasks more time that in the real sense they require. Lock (2017) explains that, project plans should factor in the concept of flexibility. In this case, the time for completion should accommodate some extra time to cater from any an unanticipated problem. For instance, unplanned system failure either hardware or software that will require ample time to fix.

Even after solving problems, there are obstacles that a programmer may face. The time limit prediction may be inaccurate and insufficient due to other predicaments that are likely to occur during planning. According to Lock (2017), new technology is way too biased in pricing than relatively older ones. Time being a factor that is limited, it may be hard to find. Moreover, it is hard to keep up with the latest trend in technology because it requires a substantial amount of investment, in this case, money and time.

In conclusion, there are several challenges that developers experience when coming up with their projects the main ones being at per with the recent technology and project time estimation. Keeping up with technology advancements requires continual learning and staying up to date with the current concepts. Excellent time management requires the breaking down of projects into manageable tasks to be able to meet project deadlines.

References

Adams, B., & McIntosh, S. (2016, March). Modern Release Engineering in A Nutshell--Why Researchers Should Care. In *Software Analysis, Evolution, and Reengineering (SANER), 2016 IEEE 23rd International Conference on* (Vol. 5, pp. 78-90). IEEE.

Berry Jaeker, J. A., & Tucker, A. L. (2016). Past the Point of Speeding Up: The Adverse Effects of Workload Saturation on Efficiency and Patient Severity. *Management Science*, *63*(4), 1042-1062.

Blake, T. (2017). Keeping Up With Technology. *Radiologic technology*, *89*(1), 86-87.

Klemp, M., Frønsdal, K. B., & Facey, K. (2011). What Principles Should Govern the Use of Managed Entry Agreements? *International journal of technology assessment in health care*, *27*(1), 77-83.

Lock, D. (2017). *The Essentials of Project Management*. Routledge.

Lye, S. Y., & Koh, J. H. L. (2014). Review on Teaching and Learning of Computational Thinking Through Programming: What is next for K-12? *Computers in Human Behavior*, *41*, 51-61.