Initial Post

◄ Initial post

Initial Post ►

Display replies in nested form

Settings

Initial Post

by Zukiswa Tuso - Monday, 5 August 2024, 2:01 PM

I have worked in various companies across the UK, primarily focusing on large and small organizations managed both locally and globally. While my direct experience hasn't been in IT, I have been involved in CRM software project implementations in many of these organizations. These implementations aimed to streamline customer management processes and improve efficiency.

However, implementing these software systems often proves challenging. My first-hand experience has shown that improper implementation, lack of user training, and system complexities frequently lead to software project failures. The most common issues I have encountered include data migration challenges, integration problems, and user adoption hurdles.

For example, one sales staff member resigned when the business migrated from a manual system to Salesforce. He refused to accept the migration, stating, "I'm a salesperson, and I don't want to be bogged down with entering data or learning a new skill on top of making sales." His resignation led to a domino effect, with many salespeople leaving the company due to similar frustrations. This is not an isolated incident; it's a common scenario in organizations attempting to improve efficiency. In 2013, Avon Products had to abandon a $125 million SAP CRM System powered by Salesforce after facing significant implementation issues. The system was too complex for sales representatives to use effectively, leading to a drop in sales and morale among the sales force (Bloomberg News, 2013).

In another organization I worked for, a similar scenario unfolded. This was their second attempt at implementing a new CRM system after the first one failed due to user difficulties, akin to Avon's experience with Salesforce. Despite hiring more salespeople, the project team struggled to train new members effectively. The data was disorganized, and ongoing issues during training sessions caused frustration. Ultimately, the company decided to switch to another CRM software, HubSpot. The project was scheduled for six months, but after this period, a new executive was hired who then fired most of the sales team and contractors.

Based on these insights, I believe the main issues contributing to software project failures are:

1. **Organization and Planning-Related Failures:**

- **Scheduling Issues:** Unrealistic schedules, management cancelling meetings, lack of accountability, and insufficient technical understanding can derail projects. The US Federal Aviation Administration (FAA) Advanced Automation System project faced extensive delays and budget overruns due to poor scheduling and procurement planning (Bar-Yam, 2003).
- **Implementing Significant Changes All at Once:** This approach often leads to delays, budget overruns, and incomplete implementation. Different roles consume information differently, so breaking projects into smaller, manageable components is crucial.

Managing the Project-Related Failures:

- **Lack of Cooperation:** Poor communication and collaboration among team members and stakeholders lead to misunderstandings, incorrect assumptions, and missed requirements. In one case study, the lack of cooperation between developers and testers resulted in significant defects being discovered late in the project cycle, causing delays and increased costs (Lehtinen et al., 2014).
- **Weak Task Backlog:** Poorly defined tasks, improper prioritization, and insufficient updates lead to inefficiencies and project delays.
- **Lack of Software Testing Resources:** Insufficient time, personnel, and proper tools result in lower-quality products and undetected defects.

Estimation Related Failures:

- **Time and Cost Estimation:** Inaccurate estimates for project duration and costs lead to significant issues. The Universal Credit system in the UK, designed to streamline welfare benefits, encountered numerous technical problems, delays, and budget overruns.

Additionally, there is often a disconnect between developers, contractors, and the organization’s sales force. Contractors may lack an understanding of the business they are working on, leading to further complications. It’s essential for all parties involved to have a clear understanding of the business and its offerings to ensure successful project outcomes.

References

- Bloomberg News, 2013. Avon Halts SAP Rollout After Orders Drop, Executives Quit. *Bloomberg*. Available at: https://www.bloomberg.com/news/articles/2013-11-14/avon-halts-sap-rollout-after-orders-drop-executives-quit [Accessed 5 August 2024].
- Bar-Yam, Y., 2003. When Systems Engineering Fails - Toward Complex Systems Engineering. *IEEE International Conference on Systems, Man and Cybernetics*. Available at: https://necsi.edu/when-systems-engineering-fails-toward-complex-systems-engineering [Accessed 5 August 2024].
- Lehtinen, T.O.A., Mäntylä, M.V., Vanhanen, J., Itkonen, J. and Lassenius, C., 2014. Perceived causes of software project failures – An analysis of their relationships. *Information and Software Technology*, 56, pp.623-643.
- Stellman, A. and Greene, J., 2005. Applied Software Project Management. 1st ed. Sebastopol, CA: O'Reilly.

Permalink Reply

Re: Initial Post

by Oi Lam Siu - Wednesday, 7 August 2024, 9:30 AM

Peer Response:

Hello Zukiswa,

Your insights into CRM software project failures are both detailed and enlightening. Your first-hand examples vividly illustrate the complexities and challenges faced during such implementations.

The Avon case you mentioned highlights the significant impact of user adoption issues, underscoring the importance of user-friendly systems. Similarly, your experience with sales staff resistance to change is a common hurdle many organizations encounter.

I also have a similar case. I remember when I started working at current company, I was informed that my role would replace the Finance Manager. She planned to retire early because she lacked confidence in learning the new ERP system and processes.

It's clear that a lack of proper planning and organization often leads to project failure. Emphasizing the need to break projects into smaller, manageable components is crucial. This approach not only helps in managing expectations but also ensures that each phase is clearly understood and executed.

In conclusion, your experiences and examples provide valuable lessons for anyone involved in software project management. Addressing scheduling issues, fostering cooperation, and focusing on realistic estimations are key strategies that can significantly improve the chances of project success. Your post serves as a practical guide for navigating the complexities of CRM implementations, and your insights are beneficial for both IT and non-IT professionals alike.

Best regards,
Helen

Maximum rating: -

Permalink Show parent Reply

◄ Initial post Initial Post ►