

The Dynamics of Implementing Software Engineering Projects

Isaac Nyabisa Oteyo^{a*}, Gregory S. Namusonge^b, Joshua Mutua^c

^{a,b,c} Jomo Kenyatta University of Agriculture and Technology, P.O. Box 62000 00200, Nairobi - Kenya

^aEmail: isaacoteyo@jkuat.ac.ke, isaacoteyo@gmail.com

^bEmail: gnamusonge@jkuat.ac.ke

^cEmail: joshua.mutua@gmail.com

Abstract

The study set out to establish whether project dynamics is perceived as a factor that affects the implementation of software engineering projects in Nairobi, Kenya and used the findings to model the dynamics of the implementation process. According to literature, the software industry in Kenya is relatively young and therefore necessary to look into other more established industries to identify key factors and challenges. Kenya's software projects experience a myriad of dynamism during implementation; the budgets are volatile, human resources required for project implementation come on board with many differing skills that add to the complexity of executing and implementing the projects, and there are also many internal and external variables to the implementation process that keep on changing. Since many projects suffer from the 90% syndrome in which a project is thought to be 90% complete for half of the total time required, software projects are not excluded. Tasks which are completed as part of a software project may be flawed and may need rework. For software projects, implementation is double phased and there are numerous changing variables in both phases that contribute to the dynamics of implementing software projects. The study used the Kenya ICT sector as a case study and utilized the findings to model the dynamics of implementing software projects. The model depicted that there is a lot of dynamism in implementing software projects. The dynamism revolves around the changing project variables that influence the success or failure of the said projects. The study recommends that modeling the dynamics of implementing any software project is critical to enable detection of any hindrances to successful implementation and avoid wastage of resources.

* Corresponding author. Tel.: +254 726 719 609

E-mail address: isaacoteyo@gmail.com; isaacoteyo@jkuat.ac.ke.

The models can aid in detecting the effects of any unforeseen uncertainties within the implementation process early enough so that appropriate action can be taken to mitigate any uncertainties.

Keywords: Software applications; project management; system dynamics; modeling; Nairobi - Kenya

1. Introduction

The development of large software systems is a complex undertaking. High cost and schedule overruns are frequent in the software development industry [1]. The recurring failures to produce large systems within planned schedule and budget have often been associated with management problems, such as ineffective communication, malformed teams, and inadequate risk analysis. Software project management is a knowledge intensive activity; therefore managers have to use their skills and experience to make decisions during the execution and implementation of software development processes.

2. ICT Usage in Kenya

According to a report by the Communication Commission of Kenya (CCK), access to ICTs has been identified by the government as a major objective of the Vision 2030 [2]. Greater access to ICT contributes towards economic growth. The report indicates that one of the key factors in the development and expansion of ICTs in Kenya has been the liberalization of the market that started in 1999. Penetration of ICTs has increased significantly over the past ten years and potential for growth is enormous since almost two thirds of the Kenyan economy corresponds to the services sector, where the use of ICTs is intensive [2].

3. Sources of Software in Kenya

Software is increasingly becoming important in the human activity [3]. Software can either be open source, proprietary or software as a service (SaaS). It is widely recognized that open source software (OSS) is freely available to anyone who needs it, but, loyalty of computer users to proprietary operating systems and general office applications seems to be still high especially in developing countries [3]. A new trend is emerging where software is hosted in the cloud and users subscribe to it as a service; meaning the physical installation of software on the users' personal computers is no longer necessary.

4. Implementation of Software Engineering Projects

The software industry has been faced with a high failure rate of projects resulting in loss of billions of financial resources [4]. Kakkar further states that as part of the solutions to this problem, the industry has to fix the software process. However, Kakkar notes that implementation as a process has not gained the expected level of importance creating a challenge of moving the software systems to the production environment.

Failure of software projects is dependent on unclear customer requirements, lack of development strategy and the outcome is transfer of blame to the human resources behind the development process. Software projects follow a life cycle that entails conception, requirements gathering, design, implementation, testing and deployment. Failure to follow the software cycle often leads to project failure [5].

The project scope, environment and implementation vary as a result of globalization, advances in technologies and deployment of software projects in varied environments [6]. The ever changing project environments pose a serious challenge to traditional approaches as a result of the resulting complexities. While implementing software projects, it is important to understand the challenges likely to be faced and put in place ideal measures to mitigate the risk of implementation failures [7].

One of the key aspects to consider while planning implementation of software projects is the project roadmap. With a roadmap it is easy to prioritize specific project components including new deployments and upgrades to existing software programs. The roadmap should be based on budget and achievable objectives. It is also important to define specific software projects through a formal and informal discovery of stakeholder needs, identifying organizational goals, challenges and opportunities. Furthermore, it is important to identify project success factors by establishing baseline measures for key project factors to assess before and after implementation. These factors include customer satisfaction, project schedule, cost, scope, and quality.

Project managers seek to deliver projects on time, on budget, and with the quality and specifications required by the customer. One of the reasons for software failures is poor quality [8]. It is apparent that poor software quality and failed software projects are creating huge costs and losses to both industry and users. Managing software engineering projects suffers from numerous problems of costing and scheduling [9]. While studying factors that influence implementation of the African Development Bank funded projects, success in any project is subject to management of a number of project constructs that include scope, project budget, project timelines and adherence to set quality standards [10].

4.1. Structuration Theory

The structuration theory, developed by Anthony Giddens in 1984, is a meta-theoretical social framework. Giddens argues that action and structure operate as a duality and simultaneously affect each other [11]. Through the structuration theory, models have been developed that make the claim that technology is constituted by human agency [12]. Agents in their actions constantly produce, reproduce and develop the social structures which both constrain and enable them [12]. Structural constraint places limits upon the feasible range of options open to an actor in a given circumstance [12].

The theory in relation to project implementation plays an important role in the assessment of the social organization of projects and the corresponding performing organizations. The application of the theory

in project management is critical to the development of a framework for project implementation responsive enough to give positive effects. The theory has been extended to adaptive structuration theory that addresses the mutual influence of technology and social processes. The adaptive structuration theory is based on among others, the proposition that social structures serve as templates for planning and accomplishing tasks. Project implementation requires adequate planning and execution of the planned implementation activities.

4.2. System Development Life Cycle Model

The dominant organizing framework for application of system development is the life cycle concept which is acknowledged as an important element in systems development [13]. This concept divides the total development cycle into identifiable stages, where each stage represents a distinct activity.

All software projects need to undergo a similar process when they are conceived, developed and implemented. Software implementation is a double phased process; implementing the functionalities of the software design (coding) and implementing the resulting product (deployment). The coding process entails converting the software specification into an executable system. Neglecting any phase of the life cycle may lead to adverse consequences. The strength behind the concept of a life cycle model lies in the creative nature of software development.

4.3. Information Systems Implementation Model

Information systems implementation models have been developed by different researchers to aid the implementation process. A model was developed for effectiveness of implementation [14]. According to the model, shown in Figure 1, the implementation process is conditioned by many factors related to the organization and the project.

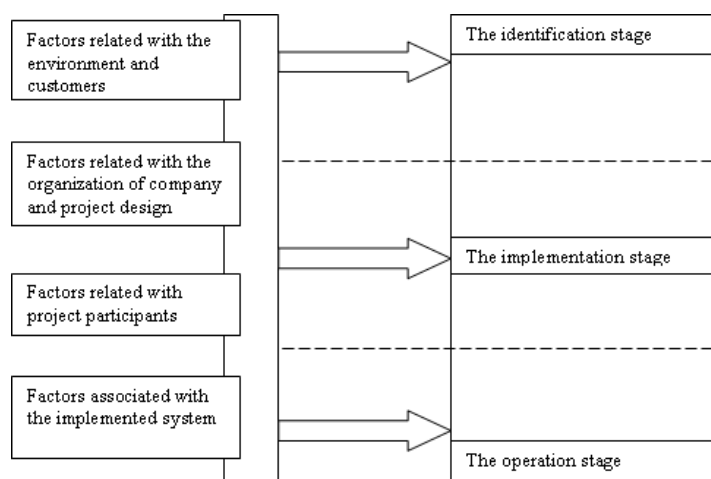


Figure 1. Reconstructed Wiechetek's Information systems implementation model

Source: Wiechetek (2012)

5. Project Dynamics

There are many complexities including dynamic behavior and feedback mechanisms as well as various interacting factors in the practical software development process [15]. The project environment is characterized by both internal and external variables [16]. Information and Communication Technology projects evolve under complex environments [17]. The complex project environments come about because of project dynamics. System dynamics approach seeks to provide understanding of complex dynamic systems over time. This is achieved through the concept of internal feed-back loops and time-delays that influence behavior in the system as a whole.

System dynamics as a method facilitates learning inside complex and non-linear systems, where the concept of both feed-back and time-delays create misperceptions [18]. In system dynamics such misperceptions can be identified and corrected if the key factors have been correctly calculated and represented. Hence, system dynamics approach allows for building and testing of policies and assumptions in order to improve understanding of system behavior or to change the observed behavior [18].

6. Results and Discussion of Findings

The objective was to establish whether project dynamics (changing project variables) was perceived as a factor that affects the implementation of software projects. To achieve this objective, a number of changing project variables were rated and the outcome is as shown in Table 1.

Table 1. Changing project variables

Pointer	Mean	Std. Deviation
Politics	3.88	1.166
Technological change	4.39	.704
Attitudes	4.52	.712
Leadership	4.36	.699
User acceptance	4.64	.699
Environment	4.00	1.000

The variables were rated on a scale of 1 - 5, where 1 was not important and 5 was very important. The mean rating was well above 4 meaning they were perceived as important changing project variables during the implementation of software projects.

Implementing projects within budget regression analysis depicts a strong relationship between changing project variables and implementing projects within budget as shown on Table 2.

Table 2. Implementing projects within budget

Model	R	R Square	Adjusted R Square	Std. Error of the Estimate
1	.974 ^a	.949	.938	.210

a. Predictors: (Constant), Environment, Politics, Technological change, Leadership, User acceptance, Attitudes

Regression analysis depicts a strong relationship between changing project variables and customer satisfaction as one of the critical areas in project implementation. The results are as indicated in Table 3.

Table 3. Customer satisfaction

Model	R	R Square	Adjusted R Square	Std. Error of the Estimate
1	1.000 ^a	1.000	1.000	.000

a. Predictors: (Constant), Environment, Politics, Technological change, Leadership, User acceptance, Attitudes

Based on the results a simulation model for the same is as shown in Figure 2. Project dynamics is perceived to have an influence on the implementation of software projects.

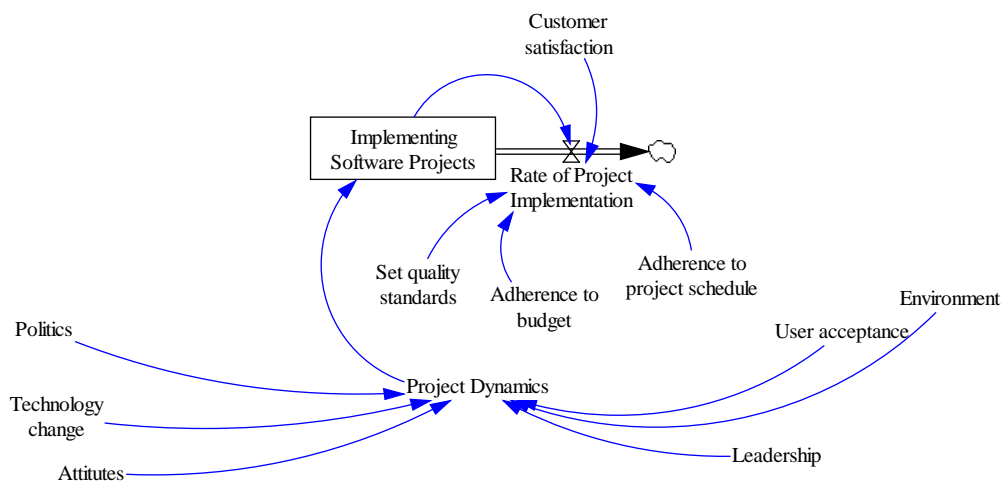


Figure 2. Project dynamics in software implementation

The model depicts that the changing project variables have a direct impact on the implementation of software projects. Based on a probability, a change in the variables can either affect the implementation process either positively or negatively. During software project implementation, financial resources

require constant monitoring, control and evaluation; the project implementation budgets have to be elastic enough to sustain the whole project implementation process. However, if the budgets are not controlled, a rise in the implementation costs will result in the project failure if the implementing firm is not liquid enough as depicted on Figure 3. This is because the rise introduces some dynamism in the implementation process that requires attention and in the process slows down the process that could lead to the eventual failure of the whole project.

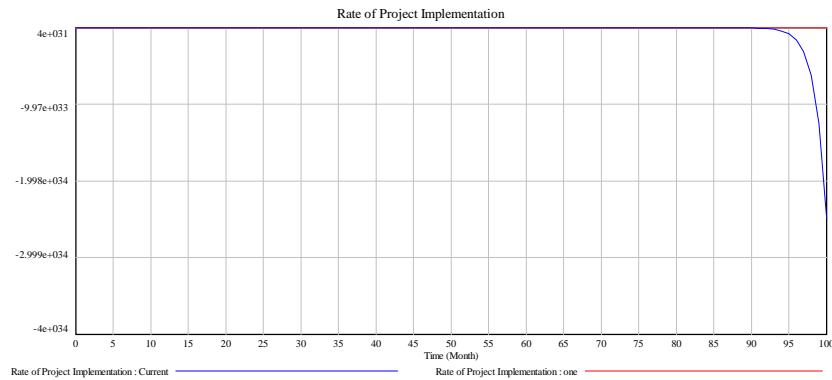


Figure 3. Implementation cost

However, if all factors remain constant without any dynamism during the implementation process, then the rate of implementation improves with time as depicted on Figure 4.

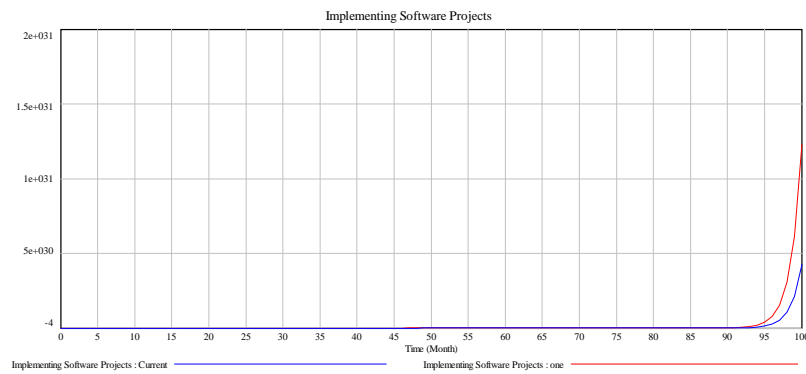


Figure 4. Constant factors during project implementation

7. Conclusion

Software projects have evolutionary growth and successful implementation of the end product is critical; set quality standards must be achieved and users must be satisfied with the end product. Any software industry has an evolutionary growth and it stabilizes with the development of quality products. The use of modeling tools for implementation of best practices reduces defects and reduces the cost to fix defects. This knowledge improves the quality and enhances the total productivity of the software development firms.

Software project implementation is double phased; coding which entails transforming the user requirements into a software product and deployment which is the actual positioning of the software product in the production environment. Both phases are influenced by almost similar factors. Modeling of these factors and their effects on the implementation process makes implementers grasp the whole process fast enough and evade any likely uncertainties that can hinder the implementation process.

While coding software projects, defect prevention and detection strategies are necessary for the development of defect-free product(s). Identification of defect at the deployment stage or even at the later stages of development is highly expensive. Defect free product has a direct and strong impact on the time, cost, and quality of the deliverables [19]. It reduces support cost, programming cost, development time, and competitive advantage.

Software development models are linear; however, agile development models have found their way into the industry. Organizations managing the software projects have their own management models as well. These structures obscure different granularity levels in task management, introduce conflict of authority and policy. This is likely to hinder effective management of, and decision-making on, quality software development process and products delivery by practitioners. There is therefore a need for developing hybrid models that encompass the software development process and management models for implementing organizations.

The study further recommends that modeling the dynamics of implementing any software project is critical to enable detection of any hindrances to successful implementation and avoid wasting resources. The models can aid in detecting the effects of any unforeseen uncertainties within the implementation process early enough so that appropriate action can be taken to evade those uncertainties.

References

- [1] M. D. O. Barros, C. M. L. Werner, P. I. Davidsen, E. Mollona, V. G. Diker, R. S. Langer, and J. I. Rowe, "Evaluating the Use of System Dynamics Models in Software Project Management," in *Paper presented at the Proceedings of the 20th International Conference of the System Dynamics Society*, 2002.
- [2] CCK, "Study on ICT Access Gaps in Kenya Report," Nairobi, 2011.
- [3] J. Kamau and S. Namuye, "A Review of Users Adoption of Open Source Software in Africa," *Comput. Inf. Sci.*, vol. 5, no. 5, pp. 45–49, Jul. 2012.
- [4] S. Kakkar, "Implementation Aspects of Software Development Projects," in *2006 Annual IEEE India Conference*, 2006, pp. 1–6.

- [5] S. Parthasarathy and N. Anbazhagan, "Significance of Software Metrics in ERP Projects," in *2006 Annual IEEE India Conference*, 2006, pp. 1–4.
- [6] R. Nienaber and E. Smith, "Adapting the SPMSA (Software Project Management Supported by Software Agents) Model to PMBOK2004 Guidelines," in *Visioning and Engineering the Knowledge Society*, 2009, vol. 5736, pp. 452–461.
- [7] G. S. Maina, "The implementation of enterprise resource planning in education sector in Kenya: case of KCA university," University of Nairobi, 2009.
- [8] C. Jones and O. Bonsignour, *The Economics of Software Quality*. USA: Addison Wesley, 2012.
- [9] J. Sterman, "System Dynamics modeling for project management. Working paper available from John Sterman," Cambridge, Mass, 1992.
- [10] J. N. Njau, "Factors influencing implementation of the African Development Bank funded project in the ministry of higher education ,science and technology - Kenya," University of Nairobi, 2012.
- [11] A. Giddens, *The Constitution of Society: Outline of the Theory of Structuration*. University of California Press, 1984.
- [12] C. M. Nyandiere, F. Kamuzora, and I. A. Lukandu, "Application of Structuration Theory and Activity Theory in Enterprise Resources Planning Systems Implementation for Universities," *Comput. Technol. Appl.*, vol. 3, pp. 385–394, 2012.
- [13] J. D. McKeen, "The Nature of Inter-Activity Relationships Within the Systems Development Cycle," Queen's University, Ontario, Canada, 1981.
- [14] Ł. Wiechetek, "Effectiveness of Information Systems Implementation: The case of the Polish Small and Medium Enterprises," in *Management, Knowledge and Learning International Conference*, 2012, pp. 193–202.
- [15] M. Wu and H. Yan, "Simulation in Software Engineering with System Dynamics : A Case Study," *J. Softw.*, vol. 4, no. 10, pp. 1127–1135, 2009.
- [16] C. K. Naimoi, "An analysis of the factors influencing successful Implementation of projects: the case of world bank funded Projects in Kenya.," University of Nairobi, 2008.
- [17] D. Gichoya, "Factors Affecting the Successful Implementation of ICT Projects in Government," *Electron. J. eGovernment*, vol. 3, no. 4, pp. 175–184, 2005.

[18] J. D. Sterman, *Business Dynamics: Systems Thinking and Modeling for a Complex World*. Chicago, USA: rwin/McGraw-Hill, 2003.

[19] T. M. Colligan, "Nine Steps to Delivering Defect-Free Software," 1997. [Online]. Available: <http://www.tenberry.com/errfree/steps.htm>. [Accessed: 20-Apr-2014].