

Chapter 3

A Triple Cornerstone Framework for Software Startups: A Systems Thinking–Based Analysis

Narendranath Shanbhag
La Trobe University, Australia

Eric Pardede
La Trobe University, Australia

ABSTRACT

Software startups are increasingly under high pressure to deliver successful products to survive and thrive in the modern highly competitive technology market. Larger organizations with deep pockets can replicate the same business ideas used by startups with relative ease. So how does the average startup stand a chance at succeeding at this seemingly David vs. Goliath contest? This chapter looks at the available literature and identifies factors that can affect the success of software development startups. Using causal loop constructs from the field of system dynamics, the interactions among the various identified factors are visualized to reveal the dynamics of the system. The result is as a three-dimensional view of success factors in form of time, capital, and (product) differentiation. The chapter also explores the cornerstones in the context of the product and business dimensions of software startups. This research acts as groundwork for analyzing the workings of software startups and sets the stage for a more holistic study upon which further research can be carried out.

1. INTRODUCTION

In recent times, software startups have disrupted the status quo and taken the market by storm with new, innovative products and services, which attempt to simplify everyday tasks in both business and personal spaces. From a product or service perspective, most software startups face stringent resource

DOI: 10.4018/978-1-7998-5788-4.ch003

constraints (Fayad et al., 2000) such as time (Paternoster et al., 2014), human resources and finances (Song et al., 2010). Additionally, startups need to be flexible and steadily evolve to stay relevant in the modern rapidly changing technology landscape (Duchesneau & Gartner, 1990). To this end, there is evidence to support the adoption of adaptable process frameworks like lean startup methodology among newer startups (Ries, 2011). There is also a good amount of interest in research directed towards startups, particularly in software development sectors. However, the software industry has some of the highest failure rates when compared to most industries. Many cases of failure can be seen in particular in early stages of the startup journey (Akkaya, 2020). Therefore, an in-depth study of the factors that can lead to better insight on factors that influence success of a product development undertaking in an aspiring software startup and would merit further study (Davis & Zweig, 2005).

Startups and the processes such as product development within it, can be considered as an example of a Socio-Economic or Socio-Technical system (Groen et al., 2008) and the complexities involved in such systems cannot be studied easily by most tools and techniques which view the problem at hand from a linear or deterministic viewpoint. According to reviewed literature, such Socio-Economic or Socio-Technical systems fall under the category of multi-loop non-linear feedback systems (Forrester, 1971). Forrester goes on to state that, as a consequence of the human evolutionary process, humans have not evolved to be good at understanding the dynamic behaviour of such complex systems. To add to this, a study found that policies and solution which have their base in linear viewpoints tend to produce solutions of a temporary nature, along with a higher number of escalating problems (Holling, 2001).

However, simply studying the various factors by isolating each one from the entire system might not suffice to provide enough insight. Studying each factor in relation to other factors present in the system can provide much better insight. As an example of this in the context of Socio-Technical systems, System Dynamics had been used to evaluate software development cycle time for effectiveness (Tvedt & Gollofello, 1995). In this study, rather than studying each factor for effectiveness in isolation, the study considered the various factors that would collectively affect the entire system. For instance, if a new software tool was introduced to a company with the intention of reducing cycle time of software development, this might reduce the amount of time the professional (using the software) needs to be involved with that task. However, the time gained by quickly accomplishing this task using the software tool might be spent as slack time by that professional. Therefore, there might be no real benefit or change to the cycle time, which was the original intention of introducing the software tool in the first place. This illustrates that a coordinated study of the various factors should be done in unison and in advance, rather than in isolation done later. By introducing an intervention such as a new software into a system and then modelling how the various elements of the system will be affected can help understand the positive and negative implications better. This is especially relevant in the case of software startups since failing to understand such implications can lead to an increase in cost, effort needed in product development and subsequently, time to market.

This work aims to group success factors for software startups, which are determined after exploring the literature and deducing various elements which affect success in software product development from the perspective of a software startup. This is done while keeping in mind aspects of dynamic behaviour which effects such Socio-Economic/Socio-Technical systems. While there are a variety of factors that lead to the success of a software startup, this would be too broad an area to study considering startups can offer a wide range of services, products, support for products/services as part of their venture. For the purposes of this study, most aspects of software startup success will be considered from a product development perspective.

29 more pages are available in the full version of this document, which may be purchased using the "Add to Cart" button on the publisher's webpage:

www.igi-global.com/chapter/a-triple-cornerstone-framework-for-software-startups/271034

Related Content

An Intelligent-Internet of Things (IoT) Outbound Logistics Knowledge Management System for Handling Temperature Sensitive Products

Joseph S.M. Yuen, K.L. Choy, H.Y. Lam and Y.P. Tsang (2018). *International Journal of Knowledge and Systems Science* (pp. 23-40).

www.irma-international.org/article/an-intelligent-internet-of-things-iot-outbound-logistics-knowledge-management-system-for-handling-temperature-sensitive-products/210114

Moth-Flame Optimization Algorithm Based Load Flow Analysis of Ill-Conditioned Power Systems

Suvabrata Mukherjee and Provas Kumar Roy (2020). *International Journal of Applied Evolutionary Computation* (pp. 1-27).

www.irma-international.org/article/moth-flame-optimization-algorithm-based-load-flow-analysis-of-ill-conditioned-power-systems/243456

Synchronization and Anti-Synchronization of Unidirectional and Bidirectional Coupled Chaotic Systems by Terminal Sliding Mode Control

Ahmad Taher Azar, Fernando E. Serrano and Nashwa Ahmad Kamal (2021). *Handbook of Research on Modeling, Analysis, and Control of Complex Systems* (pp. 399-433).

www.irma-international.org/chapter/synchronization-and-anti-synchronization-of-unidirectional-and-bidirectional-coupled-chaotic-systems-by-terminal-sliding-mode-control/271048

Structure Reliability and Response Prognostics under Uncertainty Using Bayesian Analysis and Analytical Approximations

Xuefei Guan, Jingjing He, Ratneshwar Jha and Yongming Liu (2013). *Diagnostics and Prognostics of Engineering Systems: Methods and Techniques* (pp. 358-375).

www.irma-international.org/chapter/structure-reliability-response-prognostics-under/69687

The Philosophical Roots

Lars Taxén (2010). *Using Activity Domain Theory for Managing Complex Systems* (pp. 53-64).

www.irma-international.org/chapter/philosophical-roots/39671