



The 11th International Scientific Conference
**“DEFENSE RESOURCES MANAGEMENT
IN THE 21st CENTURY”**
Braşov, November 10-11th 2016



**CURRENT RESEARCH AREAS IN DEFENSE SOFTWARE
AND INFORMATION SYSTEMS PROJECT
MANAGEMENT**

Kadir Alpaslan DEMIR*
Ebru CAYMAZ**

* Ph.D., Assistant Program Manager, Turkish Naval Research Center Command, Istanbul, Turkey

** Marmara University, Department of Organization and Management, Istanbul, Turkey

Abstract:

Scale and complexity of defense systems is increasing exponentially. Consequently, defense project schedules are long and system development costs are skyrocketing. Today, almost all large-scale defense systems are software and information technology intensive systems. Software and information systems project development in a defense context has domain-specific challenges in addition to existing challenges of large-scale system developments. Therefore, defense software and information systems project management is challenging. Current performance in many large-scale defense projects is low. Government reports list large-scale defense acquisition projects among high-risk projects. Experts point out the need for innovations on many areas of systems engineering, software engineering, and project management to successfully build large-scale systems and manage the development of these systems. According to the experts, incremental improvements will not be enough. Recent defense project experiences support the claim of these experts. Identifying the challenges and related research areas is the first step in attacking the defense acquisition problem. In this study, we identify the current major research areas in defense software and information systems project management.

Keywords: Project Management, Defense Project Management, Defense Software, Defense Information Systems, Defense Software Project Management, Defense Information Systems Project Management, Defense Acquisition, Defense Acquisition Management Research

1. Introduction

If an information system is implemented in the defense domain, then it is called a defense information system. The same logic applies to other domains as well. For example, information systems developed for healthcare industry are called healthcare information systems. The software developed as part of a defense system is called defense software. Jones defines “defense software” as “*the software developed for a uniformed military service. The term also includes software developed for the U.S. Department of Defense (DoD), or the equivalent branches in other countries.*” [1]. According to Jones, “*The main attribute that distinguishes defense software from other types of software is adherence to military or DoD standards.*” [1]. Project management subject to defense acquisition system is called defense project management.

In addition to today’s common challenges in software and information systems developments [8], there are domain specific challenges in defense software and information systems projects. US Government reports recommend improvements on information technology (IT) and weapon systems acquisitions [9, 43].

CURRENT RESEARCH AREAS IN DEFENSE SOFTWARE AND INFORMATION SYSTEMS PROJECT MANAGEMENT

Defense systems have become larger over time [2]. The complexity in weapon systems is increasing [3, 7, 11, 22]. As systems become larger and complex, development challenges increase dramatically [3]. Furthermore, as the scale of defense software increases, the project success rate drops significantly [4]. A team of experts point out that we need innovations on various aspects of systems and software engineering to overcome challenges related to development of large-scale systems [5, 6]. Incremental improvements will not be sufficient to handle these challenges. [5, 6]. Policy, acquisition, and management are listed as research areas requiring substantial new work [5]. In this study, we identify some of the current main research areas in defense software and information systems project management.

2. Current Main Research Areas in Defense Software and Information Systems Project Management

In this section, we overview the current main research areas. For each research area, we provide justifications why we should continue working on these areas.

2.1 Defense Acquisition Management

After many years of research, defense acquisitions are still costly, risky, inefficient, and full of challenges [9, 10, 43]. They are subject to heavy bureaucracy, strict regulations, and excessive oversight [11]. Defense acquisition processes are complex, inefficient, and cumbersome [32]. Furthermore, the acquisitions include many stakeholders with conflicting interests [11]. Simply, most defense acquisitions have a slow pace with low success rates. Government reports clearly state the need for improvements on defense acquisition management [12, 32]. There were many attempts to reform the defense acquisition processes [32], however most of them have failed [13]. In recent years, the US President signed “Weapon Systems Acquisition Reform Act of 2009” [14]. The bill aims for improvements both on the acquisition organization and policy. Jones recommend government officials to adopt modern civilian contracting practices [4]. Drezner points out the need for defense acquisition management improvements [7]: *“The products of the Department of Defense (DoD) acquisition process are perceived as becoming increasingly complex, emphasizing multifunction and multimission system configurations.... The management and oversight of these complex programs have similarly become more complex. Changes may be needed in the organizations and procedures used to manage the development, production, and sustainment of these complex weapon systems.”* While some stakeholders think that the US defense acquisition system is fundamentally broken, some others, including many US acquisition officials, think that the fundamental of the acquisition system is sound however improvements on the system are required [15]. One thing is for sure. Defense acquisition management is still a hot research area.

2.2 Management of Software Intensive System Developments

Today, almost all defense systems include software [11]. Most defense systems became software intensive systems [16, 17, 18, 19, 22]. In 1974, the F-16A fighter aircraft has only 135K source lines of code (SLOC). In 2012, the operational and support software of F-35 Lightning II fighter aircraft has 24 millions SLOC [19]. An Air Force General points out the importance of software in weapon systems with the following remark: *“The B-52 lived and died on the quality of its sheet metal. Today, our aircraft will live and die on the quality of software.”* [23]. Spruill states that *“Now more than ever, software is the heart of our weapons systems”* [20]. Ferguson claims that *“Software is the hidden, invisible power in weapon systems.”* [21]. Table 1 presents the increasing role of software in military aircrafts. Today, the success of software inside a defense system determines the

CURRENT RESEARCH AREAS IN DEFENSE SOFTWARE AND INFORMATION SYSTEMS PROJECT MANAGEMENT

success of the overall defense system. Furthermore, software related problems are becoming the major problems of mission-critical defense system developments [24]. Simply, defense project management is becoming software project management. As a result, defense project management is inheriting all the benefits, limitations, and challenges of software project management. Research and improvements on software project management will improve defense project management as well.

Table 1. System Functionality Requiring Software [2, 18] (Source: Partly from PM Magazine)

Defense System – Military Aircrafts	Year	% Functions Performed in Software
F-4	1960	8
A-7	1964	10
F-111	1970	20
F-15	1975	35
F-16	1982	45
B-2	1990	65
F-22	2000	80
F-35 Lightning II	2012	90

2.3 Defense System Development Process Improvements

Due to many problems faced during defense software and information systems acquisitions, US Department of Defense (DoD) funded and supported the Carnegie Mellon Software Engineering Institute (SEI) to develop Capability Maturity Models (CMMs). SEI started to work on the models during 1980s and the first CMM models were developed at the end of 1980s. Later in 2002, some of the CMM models were integrated into Capability Maturity Model Integration (CMMI). There are 3 CMMIs: CMMI for Development, CMMI for Acquisition, and CMMI for Services. Version 1.3 is the latest edition released in 2010. CMM was originally developed for software development. CMMI for development is quite applicable to organizations developing defense software and information systems, since most defense systems became software intensive systems [2]. CMMI for Development is used to assess an organization's maturity for systems development. 11 governments invest in CMMI. Organizations in 101 countries use CMMI to elevate their project development performances. Various studies show that CMMI helps to achieve better project results [1]. In the past, US DoD and various US government agencies required a certain level of CMMI certifications from contractors and subcontractors. However, this requirement is relaxed today. Since CMMI is not the only way for successful software developments. The quest for better software and system development processes is still on.

2.4 Management of Large-Scale System Developments

The scale of defense systems is constantly increasing [2]. Figure 1 shows the increasing trend of software source code size in sample major defense systems. As modern warfare is evolving, the need for capable and smarter defense systems is increasing. The advances in technology enable us to build smarter defense systems. However, development of these systems takes a long time and it is a costly effort. Furthermore, large-scale defense system developments are hard to manage. Poor performance, cost overruns, schedule overruns, and low productivity are not unusual in large-scale defense system developments. Jones report that as the defense project scale increases, the success rates

CURRENT RESEARCH AREAS IN DEFENSE SOFTWARE AND INFORMATION SYSTEMS PROJECT MANAGEMENT

decreases significantly [4]. If the software size in a defense project is around 1000 function points, the project cancellation rate is 10% [4]. When the software size is around 100K function points, then the project cancellation rate increases to 33% [4].

A study, sponsored by the US Army and conducted by an expert panel, reports that ultra-large-scale systems are the software challenge of the future [5]. Furthermore, the experts claim that “*The sheer scale of ultra-large-scale systems will change everything.*” [5]. Hayes and his colleagues point out that “*Large programs tend, by their very nature, to be (or become) very complex. The traditional tools and techniques used to manage project cost, schedule, and performance fall short when trying to manage programs in a complex environment with significant uncertainty and ambiguity.*” As a result, management of large-scale defense projects is a major research area. We need more research and studies leading to improvements and innovations in this area.

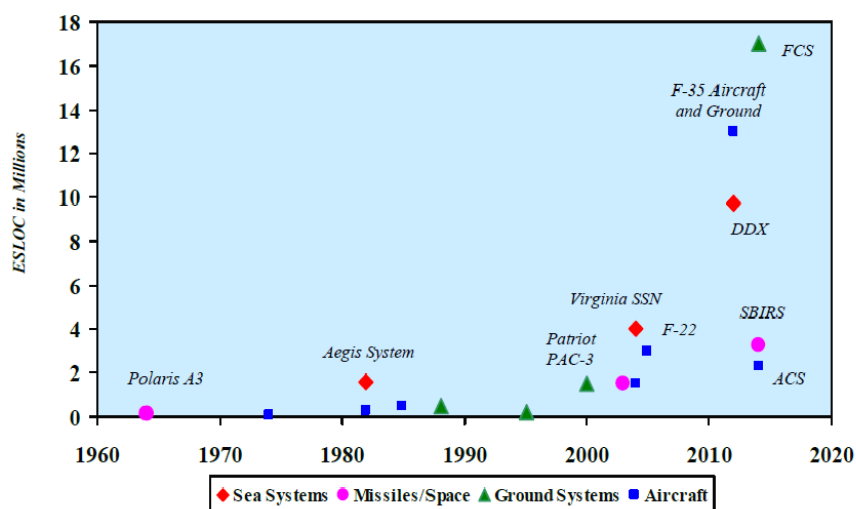


Fig.1 Software Source Code Size in Sample Major Defense Systems (Source: CARD Data, SEI, CSIS Analysis)

2.5 Implementation of Defense Enterprise Architecture Frameworks

An enterprise architecture (EA) can be defined as “*a conceptual blueprint that defines the structure and operation of an organization. The intent of an enterprise architecture is to determine how an organization can most effectively achieve its current and future objectives.*” [27]. Best composition of human and information system operations maximizing the organizational effectiveness and efficiency to achieve organizational goals is the goal of an enterprise architecture development. Furthermore, well-defined and successfully implemented enterprise architectures in federal agencies help to achieve significant cost savings [9]. Various enterprise architectures frameworks (EAFs) are developed to guide the implementation of effective enterprise architectures [28]. One of the first EAFs is Zachman Framework [29]. The Open Group’s TOGAF is developed mostly for civilian organizations. United States Federal Enterprise Architecture Framework (FEAF) is an example for EAFs developed for government organizations. There are also EAFs specially developed for military organizations. Some of these are United States Department of Defense Architecture Framework (DODAF), British Ministry of Defence Architecture Framework (MODAF), NATO Architecture Framework (NAF), Object Management Group’s Unified Architecture Framework (UAF). An overview of enterprise architecture frameworks can be found in [30].

The defense systems a military organization operates are actually the information systems of that military organization. A military organization has an enterprise architecture

CURRENT RESEARCH AREAS IN DEFENSE SOFTWARE AND INFORMATION SYSTEMS PROJECT MANAGEMENT

just like a civilian organization has. In the development of defense systems, the defense project managers should follow EAFs to satisfy the business need which is effective defense. NATO suggests the use of NAF in the development of defense systems to enable interoperability between defense information systems of partner nations. Currently, an important critic on defense EAFs is the complexity of EAFs. While what EAFs offer is valuable, the complexity of EAFs makes them hard to implement. Therefore, researches on better and simpler EAFs and how to implement them are among research areas in defense software and information systems project development. Naturally, these types of developments have various effects on defense project management.

2.6 Strategic Human Resource Development and Management for Defense Acquisition

Defense acquisition system processes are complex [25, 32]. Navigating this complex acquisition system requires both skill and experience. A project manager should have “*a versatile skill set, the ability to manage the unforeseen, and a strategic vision*” [26] to manage complex large-scale defense projects. Effective project leadership at all levels is highly correlated with project success [31]. In project management, effective management of people, process, product, and risk is essential [31]. As a result, all project members and especially project leaders at various levels should be equipped with necessary skills through education and training. In addition to the necessary knowledge and skills in software and information systems development, defense project practitioners should possess additional knowledge and skills related to project development in the defense domain. Few project managers are equipped with the necessary skills to manage complex projects [33]. There is a need for effective project managers [26, 33] and skillful engineers with the skillset necessary to successfully deliver large-scale IT projects [6]. Experts in USA draw attention to shrinking workforce in defense industry [34]. Other countries suffer from limited defense workforce, for example [35]. Defense Acquisition University [36] in USA provides a variety of education and training opportunities to defense acquisition professionals. Similar institutions exist in other countries. Strategic human resource management is also important at the organizational level [37]. Development and management of human resources both at the national and organizational level is always essential for defense industry. As defense systems evolve, we need to equip defense project managers and engineers with new skills. Search for effective ways to develop and manage human resources remain an important research area.

2.7 Systems Development Life Cycle Models

A number of systems development life cycle (SDLC) models were developed over time. Most commonly known models are waterfall, spiral, iterative, V model, and agile development. Some of these models are widely used in large-scale defense projects [2]. In fact, some of these models are developed with support or funding from defense agencies. However, the current success rates in defense projects indicate the inadequacy of these models [2]. There are various researches and case studies reporting the shortcomings of these models [2]. For example, the waterfall model is insufficient to handle frequent requirement changes during system development. Furthermore, if the project involves a certain amount of R&D, then the project management is quite problematic with the waterfall model. However, the waterfall model is quite compatible with defense acquisition processes. Spiral and iterative models are better in handling requirements changes and risk management. However, they are not fully compatible with many current acquisition processes and strict government oversight procedures. The high productivity achieved with agile development models draws the attention of defense community. The discussions

CURRENT RESEARCH AREAS IN DEFENSE SOFTWARE AND INFORMATION SYSTEMS PROJECT MANAGEMENT

related to agile development models deserve a separate section. Therefore, the next section presents these discussions.

Boehm and his colleagues recently developed incremental commitment spiral model (ICSM) for systems and software development [38]. Boehm and Lane claim that the model is applicable to defense projects [39]. In a prior study, we list the requirements of systems development life cycle models for large-scale defense systems [2]. We also point out the need for better models and outline a research agenda.

2.8 Adoption of Agile Software Development Practices in Defense Projects

Agile software development may be considered as a specific type of systems development life cycle model. There are various models that fall under the category of agile software development methodologies. Some of the agile models are adaptive software development, extreme programming, and Scrum. Basically, models following the agile manifesto are called agile development models. There are 12 principles behind the agile manifesto [40]. Customer satisfaction with quick and continuous delivery is the most distinguishable characteristic of agile models. They are developed in response to the increased competition in software and IT industry. Originally agile models are designed to handle small to medium scale software projects. High productivity achieved using these models attracts the attention of large-scale system developers. However, certain principles of agile manifesto are hard to apply in large-scale defense system development projects. For example, short face to face meetings with project team members are important in agile methodology. Such principle is hard to follow in a large project requiring tens or hundreds of developers. The applicability of agile software development to defense projects and how to successfully adopt agile development practices are hot research areas for defense project management.

3. Conclusion

We need innovations on many aspects of systems engineering and related disciplines to overcome the challenges brought by the increasing scale in IT systems [6, 26]. Various experts point out that incremental improvements to existing practices and tools will not be enough to successfully manage large-scale IT system development [6, 26]. Furthermore, innovations on project management of large-scale system developments are required. Defense context adds further challenges. Today, almost all large-scale defense systems are software intensive systems. They heavily rely on information technology. Therefore, defense project management has become software or information systems project management. A defense project manager has to overcome both the challenges related to software and information systems project management and the challenges related to defense domain.

In this study, we identified the current major research areas of defense software and information system project management. While the research areas identified are not complete, we briefly discussed the most essential ones. This list of research areas is a starting point for further research. Future work may include identification of other research areas. Another line of future work may be root cause analyses of defense project management challenges.

Acknowledgements and Disclaimers:

The views and conclusions contained herein are those of the authors and should not be interpreted as necessarily representing the official policies or endorsements, either expressed or implied, of any affiliated organization or government.

CURRENT RESEARCH AREAS IN DEFENSE SOFTWARE AND INFORMATION SYSTEMS PROJECT MANAGEMENT

References:

- [1] Jones, C., (2002). Defense Software Development in Evolution, Crosstalk-The Journal of Defense Software Engineering, November 2002.
- [2] Demir, K. A., (2015). Requirements for Systems Development Life Cycle Models for Large-Scale Defense Systems. Journal of Defense Resources Management (JoDRM), 6(2), 87-94. http://journal.dresmara.ro/issues/volume6_issue2/11_demir.pdf, accessed on October 10, 2016.
- [3] Naval Research Advisory Committee, (2006). Software Intensive Systems, http://www.nrac.navy.mil/docs/2006_rpt_software_intensive_systems.pdf, accessed on October 10, 2016.
- [4] Jones, C., (2000). Software assessments, benchmarks, and best practices. Addison-Wesley Longman Publishing Co., Inc., 2000.
- [5] Northrop, L. et al., (2006), Ultra-large-scale systems: The software challenge of the future. Carnegie-Mellon University, Pittsburgh, PA, USA. Software Engineering Institute (SEI). http://www.sei.cmu.edu/library/assets/ULS_Book20062.pdf , accessed on October 10, 2016.
- [6] Sommerville, I., Cliff, D., Calinescu, R., Keen, J., Kelly, T., Kwiatkowska, M., McDermid, J., Paige, R., (2012). Large-scale complex IT systems. Communications of the ACM, 55(7), 71-77.
- [7] Drezner, J. A., (2009). Competition and Innovation under Complexity. Santa Monica, CA: RAND Corporation, 2009. <http://www.rand.org/pubs/reprints/RP1386.html> , accessed on October 10, 2016.
- [8] Demir, K. A., (2009). A Survey on Challenges of Software Project Management. In Software Engineering Research and Practice (pp. 579-585).
- [9] United States Government Accountability Office, Report to Congressional Committees: 2015 Annual Report: Additional Opportunities to Reduce Fragmentation, Overlap, and Duplication and Achieve Other Financial Benefits, April 2015, Report No: GAO-15-404SP, <http://gao.gov/assets/670/669613.pdf>, accessed on October 10, 2016.
- [10] United States Government Accountability Office, Report to Congressional Committees: Acquisition Reform - DOD Should Streamline Its Decision-Making Process for Weapon Systems to Reduce Inefficiencies, February 2015, Report No: GAO-15-192, <http://www.gao.gov/assets/670/668629.pdf>, accessed on October 10, 2016.
- [11] Demir, K. A., (2009). Challenges of weapon systems software development. *Journal of Naval Science and Engineering*, 2009, 5(3). pp. 104-116. (*Deniz Bilimleri ve Mühendisliği Dergisi*, in Turkish).
- [12] United States Government Accountability Office, Defense Acquisition Process, Military Service Chiefs' Concerns Reflect Need to Better Define Requirements before Programs Start, June 2015, Report No: GAO-15-469, <http://www.gao.gov/assets/670/668629.pdf>, accessed on October 10, 2016.
- [13] Steinbock, D., (2014). The Challenges for America's Defense Innovation. The Information Technology and Innovation Foundation (ITIF). <http://www2.itif.org/2014-defense-rd.pdf>, accessed on October 10, 2016.
- [14] United States, Weapon Systems Acquisition Reform Act of 2009, <http://www.govtrack.us/congress/billtext.xpd?bill=s111-454>, accessed on October 10, 2016.
- [15] Hunter, A., (2015). The Battle over How to Save Acquisition, Center for Strategic & International Studies (CSIS) Report, November 16, 2015, <https://www.csis.org/analysis/battle-over-how-save-acquisition> , accessed on October 10, 2016.

CURRENT RESEARCH AREAS IN DEFENSE SOFTWARE AND INFORMATION SYSTEMS PROJECT MANAGEMENT

- [16] Goldin, L., Matalon-Beck, M., Lapid-Maoz, J., (2010). Reuse of requirements reduces time to market. In Proceedings of 2010 IEEE International Conference on Software Science, Technology and Engineering (SWSTE), pp. 55-60. 15-16 June 2010, Herzlia, Israel. <http://thegoldensolution.com/wp-content/pdfs/12.pdf> doi: 10.1109/SwSTE.2010.17.
- [17] Nelson, M., Clark, J., Spurlock, M. A., (1999). Curing the software requirements and cost estimating blues, PM Magazine, November-December 1999, pp. 54-60.
- [18] Report of the Defense Science Board Task Force on Defense Software, (2000). Defense Science Board, November 2000, pp. 11. <http://www.acq.osd.mil/dsb/reports/ADA385923.pdf> , accessed on October 10, 2016.
- [19] Hagen, C., Sorensen, J., (2013). Delivering Military Systems Affordably, Defense AT&L, March-April 2013, accessed on October 10, 2016. http://www.dau.mil/pubscats/ATL%20Docs/Mar_Apr_2013/Hagen_Sorenson.pdf
- [20] Spruill, N., (2002). Now more than ever, software is the heart of our weapons systems, Crosstalk - The Journal of Defense Software Engineering, January 2002, <http://www.crosstalkonline.org/storage/issue-archives/2002/200201/200201-Spruill.pdf>, accessed on October 10, 2016.
- [21] Ferguson, J., (2001). Crouching dragon, hidden software: software in DoD weapon systems. IEEE Software, 18(4), 105.
- [22] Demir, K. A., (2005). Analysis of TLcharts for weapon systems software development. Master's Thesis, December 2005, Naval Postgraduate School, Monterey, CA. http://calhoun.nps.edu/bitstream/handle/10945/1825/05Dec_Demir.pdf?sequence=1, accessed on October 10, 2016.
- [23] Hagen, C., Sorensen, J., Hurt, S., Wall, D., (2012). Software: The Brains behind U.S. Defense Systems, A.T. Kearney Inc., 2012. https://www.atkearney.com/documents/10192/247932/Software-The_Brains_Behind_US_Defense_Systems.pdf/69129873-eccc-4ddc-b798-c198a8ff1026 , accessed on October 10, 2016.
- [24] U.S. Government Accountability Office, Mission-Critical Systems – Defense Attempting to Address Major Software Challenges, GAO/IMTEC-93-13, December 1992. <http://www.gao.gov/assets/220/217352.pdf> , accessed on October 10, 2016.
- [25] Performance of the Defense Acquisition System, 2015 Annual Report. Washington, DC: Under Secretary of Defense, Acquisition, Technology, and Logistics (USD[AT&L]), September, 2015. <http://www.acq.osd.mil/fo/docs/Performance-of-Defense-Acquisition-System-2015.pdf> , accessed on October 10, 2016.
- [26] Meier, S. R., (2013). Leading Complex Projects in the DoD, Defense AT&L, May – June, 2013. http://www.au.af.mil/au/awc/awcgate/dau/meier_mj13.pdf, accessed on October 10, 2016.
- [27] <http://searchcio.techtarget.com/definition/enterprise-architecture>, accessed on October 10, 2016.
- [28] Demir, K. A., (2015). Multi-View Software Architecture Design: Case Study of a Mission-Critical Defense System. Computer and Information Science, 8(4), 12. <http://www.ccsenet.org/journal/index.php/cis/article/view/51025/28835> , accessed on October 10, 2016.
- [29] Zachman, J., (1987). A framework for information systems architecture. IBM Systems Journal, 1987, 26(3), 276-292. doi: 10.1147/sj.263.0276.
- [30] Reichwein, A., Paredis, C. J., (2011). Overview of architecture frameworks and modeling languages for model-based systems engineering. In Proceedings of the ASME 2011 International Design Engineering Technical Conferences and Computers and Information in Engineering Conference, pp. 1341-1349.

CURRENT RESEARCH AREAS IN DEFENSE SOFTWARE AND INFORMATION SYSTEMS PROJECT MANAGEMENT

- [31] Demir, K. A., (2008). Measurement of software project management effectiveness. Doctoral Dissertation, Naval Postgraduate School, Monterey, CA, USA. December 2008.
- [32] Schwartz, M., (2014). Defense Acquisitions: How DOD Acquires Weapon Systems and Recent Efforts to Reform the Process, Congressional Research Service (CRS) Report, <https://www.fas.org/sgp/crs/natsec/RL34026.pdf> , accessed on October 10, 2016.
- [33] Hayes, S., Kopunic, D., Wood, R., (2011). Program Complexity: Can We Understand It? Can We Overcome It? Defense AT&L: May–June 2011. http://www.dau.mil/pubscats/ATL%20Docs/May-June11/Hays_Kopunic_Wood.pdf , accessed on October 10, 2016.
- [34] Zillman, C., (2013). America's defense industry is going gray. Fortune, November 12, 2013, <http://fortune.com/2013/11/12/americas-defense-industry-is-going-gray/>, accessed on October 10, 2016.
- [35] Caymaz, E., Erenel, F., (2012). Addressing the Need of Qualified Personnel in Defense Industry: The Turkish Case. The 7th International Scientific Conference “Defense Resources Management in the 21st Century”, Braşov, November 15, 2012. pp. 25-29.
- [36] <http://www.dau.mil/default.aspx>, accessed on October 10, 2016.
- [37] Demir, K. A., (2016). Strategic Human Resource Management of Government Defense R&D Organizations. Crosstalk-The Journal of Defense Software Engineering, 29 (2), 24-30. http://www.softwaresuccess.org/papers/2016_Demir_Crosstalk_201603.pdf , accessed on October 10, 2016.
- [38] Boehm, B., Lane, J. A., Koolmanojwong, S., Turner, R., (2014). The incremental commitment spiral model: Principles and practices for successful systems and software. Addison-Wesley Professional.
- [39] Boehm, B., Lane, J. A., (2007). Using the incremental commitment model to integrate system acquisition, systems engineering, and software engineering. Crosstalk- The Journal of Defense Software Engineering, 2007, 19(10), 4-9.
- [40] <https://www.agilealliance.org/agile101/12-principles-behind-the-agile-manifesto/> , accessed on October 10, 2016.
- [41] Demir, A., Caymaz, E., Erenel, F., (2016). Defense Industry Clusters in Turkey. Journal of Defense Resources Management (JoDRM), 1(7), 7-20. http://journal.dresmara.ro/issues/volume7_issue1/01_demir_caymaz_erenel.pdf , accessed on October 10, 2016.
- [42] Erenel, F., Demir, K. A., Caymaz, E., (2015) Assessment Of Defense Industry Clusters In Turkey. Defense Resources Management in the 21 St Century, 13 November 2015, Braşov, Romania, http://conference.dresmara.ro/conferences/2015/CoDRM_2015.pdf#page=101 , accessed on October 10, 2016.
- [43] U.S. Government Accountability Office, Report to Congressional Committees: High Risk Series, Report No: GAO-15-290, February 2015. <http://www.gao.gov/assets/670/668415.pdf> , accessed on October 10, 2016.