ASSESSING APPLICATION LIFECYCLE MANAGEMENT POTENTIALS: AN EXPLORATORY INDUSTRIAL CASE STUDY

A THESIS SUBMITTED TO
THE GRADUATE SCHOOL OF NATURAL AND APPLIED
SCIENCES OF
ÇANKAYA UNIVERSITY

BY

ZÜLEYHA AKGÜN

IN PARTIAL FULFILLMENT OF THE REQUIREMENTS FOR THE
DEGREE OF
MASTER OF SCIENCE
IN
THE DEPARTMENT OF
COMPUTER ENGINEERING

JANUARY 2019

Title of the Thesis: Assessing Application Lifecycle Management Potentials: An Exploratory Industrial Case Study

Submitted by **Züleyha AKGÜN**

Approval of the Graduate School of Natural and Applied Sciences, Çankaya University

Prof. Dr. Can ÇOĞUN
Director

I certify that this thesis satisfies all the requirements as a thesis for the degree of Master of Science

Prof. Dr. Erdoğan DOĞDU
Head of Department

This is to certify that I have read this thesis and that in my opinion it is fully adequate, in scope and quality, as a thesis for the degree of Master of Science.

Dr. Instructor Murat YILMAZ
Supervisor

Examination Date: 22 January 2019

Examining Committee Members

Prof. Dr. Alok MISHRA (Atılım Univ.)

Associate Professor M. Alp ERTEM (Çankaya Univ.)

Dr. Instructor Murat YILMAZ (Çankaya Univ.)

STATEMENT OF NON-PLAGIARISM PAGE

I hereby declare that all information in this document has been obtained and presented in accordance with academic rules and ethical conduct. I also declare that, as required by these rules and conduct, I have fully cited and referenced all material and results that are not original to this work.

Name, Last Name : **ZÜLEYHA AKGÜN**

Signature :

Date : 8 February 2019

ABSTRACT

ASSESSING APPLICATION LIFECYCLE MANAGEMENT POTENTIALS: AN EXPLORATORY INDUSTRIAL CASE STUDY

AKGÜN, Züleyha

M.Sc., Department of Computer Engineering

Supervisor: Dr. Instructor Murat YILMAZ

January 2019

Software projects require a good management system to achieve a successful result because they are risky and often spread over a long period. The most important element of a good management is controlled planning. Within the scope of the project, all necessary stages of the software development life cycle need to be accurately and consistently planned and documented to complete the requirements, analysis, design, development process, estimations, risk and change control methods, test and maintenance processes in a complete and accurate manner.

The notion of software lifecycle management requires the incorporation of effective software engineering processes that encompass not only development but also deployment and maintenance. Although new methodologies for software engineering continue to emerge, still software development continues to experience a significant amount of failure rates. To overcome the complexities of modern software development, agile development philosophy suggests that software developers should remain involved with the application and working continuously with stakeholders to make required changes incrementally by tracking application changes. Application Lifecycle Management (ALM) is an emerging trend that reflects a novel attitude towards software development also expressed as the DevOps, which blends the tasks performed by a firm's software development and systems operations teams, monitor software artifacts from inception to completion while changes are noted.

The ultimate outcome from this thesis is an assessment framework for delivering ALM as a service to a set of selected software projects in a large-sized software development organization. It should guide practitioners to tailor ALM practices from the idea phase

through the selection of the underlying tool set. Finally, an industrial case study is conducted to assess the usability of this approach. This study guides ALM transformation using agile best practices in a large-scale corporate environment. In light of these, a unified ALM toolset and processes will be discussed with the economic implications of the transformation process and the lessons learned.

Keywords: Application Life-Cycle Management, Software Development Life-Cycle Management, Software Process, Application Life-Cycle Management Tools, Action Research, Industrial Case Study.

ÖZ

UYGULAMA YAŞAM DÖNGÜSÜ YÖNETİMİ POTANSİYELLERİNİN DEĞERLENDİRİLMESİ: ENDÜSTRİYEL ARAŞTIRMA ÇALIŞMASI

AKGÜN, Züleyha
M.Sc., Bilgisayar Mühendisliği Bölümü
Tez Yöneticisi: Dr. Ögr. Üyesi Murat YILMAZ
Ocak 2019

Yazılım projeleri, risk barındıran ve genellikle uzun bir sürece yayılmış olduğundan, başarılı bir sonuç elde etmek adına iyi bir yönetim sistemi gerektirir. İyi bir yönetimin en önemli unsuru ise kontrollü planlamadır. Proje kapsamında, gereksinimler, analiz, tasarım, geliştirme süreci, kestirimler, risk ve değişiklik kontrol yöntemleri, test ve bakım süreçlerinin tam ve doğru bir şekilde tamamlanması için yazılım geliştirme yaşam döngüsünün gerekli tüm aşamalarının doğru ve kararlı bir şekilde planlanması ve sistematik içerisinde yürütülmesi gerekir.

Yazılım yaşam döngüsü yönetimi kavramı, yalnızca gelişmeyi değil, aynı zamanda dağıtım ve bakımı da kapsayan etkili yazılım mühendisliği süreçlerinin dâhil edilmesini gerektirir. Yazılım mühendisliği için yeni metodolojiler ortaya çıkmaya devam etse de, yazılım geliştirme hala önemli miktarda başarısızlık oranını deneyimlemeye devam etmektedir. Modern yazılım geliştirmenin karmaşıklıklarının üstesinden gelmek için, çevik kalkınma felsefesi, yazılım geliştiricilerinin uygulama ile ilgili olarak kalmasını ve uygulama değişikliklerini izleyerek gerekli değişiklikleri aşamalı olarak yapmak için paydaşlarla sürekli olarak çalışmasını önermektedir. Uygulama Yaşam Döngüsü Yönetimi (ALM), bir firmanın yazılım geliştirme ve sistem operasyon ekipleri tarafından gerçekleştirilen görevleri harmanlayan, yazılım girişimlerini başlangıçtan sonuçlanana kadar tamamlayan ve değişikliklerin kaydedildiği DevOps olarak ifade edilen yazılım geliştirmeye yönelik yeni bir tavrı yansıtan yeni bir eğilimdir.

Bu tezin nihai sonucu, büyük ölçekli bir yazılım geliştirme organizasyonunda ALM'nin bir dizi seçilmiş yazılım projesine hizmet olarak sunulması için bir değerlendirme çerçevesidir. Uygulayıcıları, ALM uygulamalarını fikir aşamasından temel araç setinin seçimiyle uyarlamaya yönlendirmelidir. Son olarak, bu yaklaşımın kullanılabilirliğini değerlendirmek için endüstriyel bir vaka çalışması yapılmıştır. Bu çalışma büyük ölçekli bir şirket ortamında çevik en iyi uygulamaları kullanarak ALM dönüşümüne rehberlik etmektedir. Bunların ışığında, birleşik bir ALM araç seti ve süreçleri, dönüşüm sürecinin ekonomik sonuçları ve öğrenilen dersler ile tartışılacaktır.

Anahtar Kelimeler*:* Uygulama Yaşam Döngüsü, Yazılım Geliştirme Yaşam Döngü Yönetimi, Yazılım Süreçleri, Uygulama Yaşam Döngü Yönetim Araçları, Eylem Araştırması, Endüstriyel Çalışma.

ACKNOWLEDGEMENTS

I would like to express the best regards to Genel Manager of AKGÜN SOFTWARE, Temel AKGÜN, whom supports me and ALM process improvement to complete this work within the company.

This work has been carried out at AKGÜN SOFTWARE. We would like to thank all members of AKGÜN SOFTWARE developer productivity tools team.

I would like to express my sincere gratitude to my thesis advisor, Dr. Murat YILMAZ. He is one of the best supervisors, who has encouraged and guided me throughout this thesis patiently. I had a lot of useful discussions with him. I would like to thank him cordially for his valuable and excellent comments.

I would like to thank and express my deep gratitude to my father, Temel AKGÜN, whom we have shared good and bad times for years. He is such a a great father in all over the world. He always believes me that I can complete this study, encouraged and supported me to complete my thesis in a quality way. I could not complete this study without him and his limitless and endless support, reliance and guidance. Also, I would like to thank to my mother, Samiye AKGÜN, who has encouraged me. Thank you once again to my family for making feel me that I have a great family. My family is so special for me that they always support me for my whole education life.

Finally, I would like to thank to my spouse. He always believes me that I can complete this study. In stressful situations, he understood me and has worked with me in a coordination.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF SCREEN SHOTS

LIST OF ABBREVIATIONS

ALM	Application Lifecycle Management
PLM	Product Lifecycle Management
SDLC	Software Development Lifecycle Management
QA	Qality Assurance
POC	A Proof of Concept
IT	Information Technology
RQ	Research Questions
HP	Hewlett-Packard
TFS	Team Foundation Server
IBM	International Business Machines
ET	Enterprise Tester
MS	Microsoft
RRC	Rational Requirement Management
RQM	Rational Quality Management
RTC	Rational Team Concert
RCCM	Rational Change and Configuration Management
LDAP	Lightweight Directory Access Protocol
PC	Personal Computer
CMMI	Capability Maturity Model Integration
QTP	Quick Test Professional

CHAPTER 1

1. INTRODUCTION

Software projects are complex structures that contain many scopes and are carried out on the basis of a regular process. This process starts with gathering the requirements and continues until reaching the end-user. The process includes in itself the stages of planning, gathering, analysis of requirements, design, production, testing, maintenance and distribution. It is the most important and required point during the negotiations with the stakeholder to have a clear knowledge of the project scope, quality, cost and schedule elements. One of the most important points of the software life cycle that is carried out within the scope of software projects is the planning, which is the first step of the process. By means of appropriate and accurate planning, while necessary controls are made to prevent the project from deviating from the intended objectives and measures are taken, it also allows the customer to have a clear idea about the project and it is easier for the technical team to predict which results will be encountered at which stages of the project. In this way, many measures are taken to ensure the success of the project. Even though Software Development Life Cycle (SDLC) and Application Life-Cycle Management (ALM) seem to be similar concepts, Application Life-Cycle Management (ALM) has a wider scope. While the Software Development Life Cycle includes the software methodologies, the ALM is an ongoing process that manages the life (growth, development, maintenance) of an application. Application Life-Cycle Management can be considered as a combination of business management and software engineering. In projects, the most important element of the correct implementation of the software life cycle is through good management. Application Life-Cycle Management (ALM) is a solution that manages all of the processes from development to complete disuse. At the same time, this solution gathers together and automates the applications running on different platforms on a single platform and controls the costs by reducing the labor requirements. The importance of these tools in project development processes is increasing with the developments in ALM tools. In all processes of a project from the idea stage to the

delivery and maintenance phase, ALM tools have a significant role [3]. Therefore, the increase in the efficiency of the ALM tools will increase the quality of the product to be obtained from the project and will save time as well as the project costs. The most important way of increasing the output from the ALM tools is to increase the knowledge and experience of those using these tools and working in different areas (software engineers, test engineers, analysts, configuration managers, project managers, etc.) in ALM processes. Various trainings and seminars can be organized to increase the knowledge and experience of the employees in ALM tools. To use the Application Life-Cycle Management tools correctly and to integrate them into software projects is a serious effort. A long-term, well-prepared reporting system brings success. One of the biggest gains of a properly-functioning Application Life-Cycle Management is the detailed reporting of the whole process.

Application Life-Cycle Management Structure:

Application Life-Cycle Management (ALM) is an approach covering all processes of a software project output from idea to development, distribution and maintenance [1]. This provides a powerful reporting and traceability capability throughout the software development period [5]. An application needs to be developed in the first place and then to be customized in accordance with the feedback from the customer and maintained to exist for a long time. There is a need for an ongoing process for all these works. The ALM is a concept that meets this requirement. In fact, the main purpose here is to improve the software quality and to make efforts to reduce the costs [4]. ALM includes some very important building blocks within. For example; it has methodologies that enable the management of such concepts as the management of claims, software architecture, design, coding, testing of the product, review of the live processes, product improvement as a result of the review, releasing new versions as a result of these improvements [6].

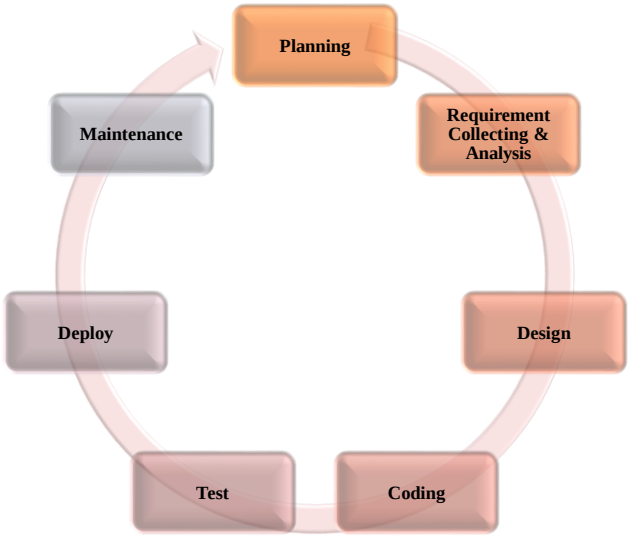


Figure 1: Application Lifecycle

In addition, ALM supports the life of an application with tool and process.

Tool: Enables that individuals are more productive so that the product can be released in a faster way and at an earlier time.

Process: Provides methodologies to be followed for a product of higher quality.

Corporate companies work in accordance with various standards for more systematic and qualified progress of their software projects [4]. These standards are very important both in terms of customer satisfaction and the general reputation of the company. As reporting is one of biggest gains of the ALM system, it is very important to use the ALM system in an efficient way to document and improve the standards. The ALM concept has emerged with the version name of 1.0. The ALM 1.0 suggests the use of different tools for each discipline and compatible functioning of these tools between themselves as a solution.

Different tools that include different ALM processes such as planning, development, design, testing, maintenance etc., may exchange information and interoperate with the tools that manage other processes [3]. Due to the problems and shortcomings in the integration of the different tools manufactured by different firms in ALM 1.0 solution with one another, the ALM 2.0 solution emerged [5,6]. In the ALM 2.0 approach, the software processes are gathered under a single tool rather than following the processes by means of different tools and the areas different from the processes [9, 14]. Tools

suitable for the ALM 2.0 solution follow a role and authorization-based approach throughout this entire process. Data processing costs in ALM 1.0 solution are set to zero as data processing between the processes in ALM 2.0 version are automatically provided on a regular basis. In this way product output production, software development management and reporting are performed more quickly [7,8]. The ALM approach covers all processes of requirements management, test management, building management, source code management, project management and risk management. Carrying out all these different processes as integrated with each other is very important in successful conclusion of a software project and in determining the quality level.

Gathering the concepts of communication, cooperation and integration under a single title in the Information Technologies (IT) has recently been achieved with the DevOps. DevOps concept is formed by the combination of Development and Operations. "*Dev*" is usually used for the software developers but in fact it represents a wider community and includes everyone working for software development. "*Ops*" has a broader scope compared to "*Dev*" and is used for system engineers, system administrators, release engineers, database administrators (DBAs), network engineers, security experts and for many more different sub-disciplines. Application Life-Cycle Management (ALM) and DevOps solutions help to meet business needs while improving software quality and developer productivity. The ALM automates the entire software development and delivery process from the receipt date of request by means of a distribution way. The system documents and monitors everything that occurs automatically, creating valuable data to enable continuous process improvement. Comprehensive and real-time information on ALM data allows DevOps and IT teams to collaborate in providing better software and services to the users.

Application Life-Cycle Management is based on 3 fundamental structures: Traceability, Automation and Reporting [1]. Thereby, instead of limiting application lifecycle management to software development, it enables the business units, business analysts, corporate architects, software developers, test teams, operating teams and executives of all these teams to monitor and manage it on a shared platform. In this way, it is possible to conduct an accurate study by including the opinions of all staff,

to whom the software will serve, and ensuring that the software developed fully meets the needs of the company.

There are many ALM tools used in our day. These are: IBM Rational Team Concert (IBM JAZZ), Microsoft Team Foundation Server (TFS), HP Quality Server, Atlassian JIRA etc. The three main ALM tools used by the corporate companies of our day are compared in the following table:

Table 1: 3(Three) Main ALM Tools

TOOL / PROPERTY	**Atlassian JIRA**	**Rational Team Concert (IBM JAZZ)**	**Microsoft Team Foundation Server**
DEFINITION	Patient monitoring and project planning tool. Used as cloud service and commercial tool.	The project management component of the IBM JAZZ Platform.	It is a source code management system that provides project management features.
CATEGORY	Project Planning Issue Management	Project Planning	Project Planning
SUPPLIER	Atlassian	IBM JAZZ	Microsoft
THE FIRST VERSION	2002	2008	2005
OPERATING SYSTEMS	Linux Solaris Windows	Adk Ibm Ibm Z/Os Lınux Solaris Windows	Windows
MOBILE APPLICATION	Android IOS	-	Android IOS Windows Phone
IDENTITY VERIFICATION	Password LDAP	Password LDAP	Password http Verification Active Directory

NOTIFICATION	E-mail	E-mail	E-mail
HIERARCHICAL TASKS	Yes	Yes	Yes
GANTT CHART	Yes	Yes	-
RESOURCE MANAGEMENT	-	Yes	Yes
TIME MONITORING SYSTEM	Yes	Yes	-
RİSK MANAGEMENT	Yes	Yes	Yes
KANBAN SUPPORT	Yes	Yes	Yes
SCRUM SUPPORT	Yes	Yes	Yes

*It is best for the corporate companies to choose ALM Tool based on the project requirements. When the table is examined in detail, it is seen that a tool with all the features has not been released yet. With various version changes, completion of missing features is taken into account by the developers.

The specific objective of this study was to automatize the production benches of AKGUN YAZILIM, to ensure that the ALM methodologies are implemented in AKGUN YAZILIM and that the production lines run in an integrated way during the product lifecycle, to digitalize the processes and to enable ALM transformation.
This study aimed to address the following research question: Can we orchestrate a large-size software development organization by using ALM transformation process and agile best practices?

The overall structure of the study takes the form of five chapters; including this introduction chapter. This thesis is structured as follows:

Chapter 2 begins by reviewing the literature of the research and gives ALM details.

Chapter 3 describes based on the research techniques, which are used in our study. Then, an action research design used in this study is illustrated.

Chapter 4 details the action research. Application Lifecycle Management Action Research consists of three phases, which are ALM 1.0, ALM 2.0 and ALM 3.0. All action research phases are detailed in these topics: Diagnosis, Action Planning, Action Taking, Evaluating, Specifying the Learning.

Chapter 5 summarizes the overall study and discusses the implications of the answers of the research question. Then, the conclusion of the study is explained in detailed. And, chapter 5 explains what we will do for this study in the future.

CHAPTER 2

2. BACKGROUND AND RELATED WORKS

2.1. Introduction

There is little academic research and published paper on this topic, the next paragraphs outline the literature details. By clarifying the ALM concept, its importance and place in the projects will be detailed and its advantages will be emphasized by providing information on at which stages of the projects of the corporate companies it is used.

In addition, Application Life-Cycle Management is explained in this section and information on Application Life-Cycle Management and Software Life-Cycle Management is given and details are given for the Application Life-Cycle Tools, which take an active role in the management of software projects of our day. The Application Life-Cycle Management and its place and importance in the software projects of our day are addressed in detail and the solution offered for the problems in the articles by using the Application Life-Cycle Management tools is explained in depth.

2.2. Application Lifecycle Management Description and Scope

Herden et al. [10] use the definition of "*Application Life-Cycle Management (ALM) is the set of processes, methods and tools used by the IT departments of an organization to ensure the smooth operation of the critical business practices and adaptation to new requirements*" and enable that the concept is better understood. They argue that for large-scale business practices, the number of documents to be referred and manual actions to be performed needs to be minimized to allow agile and optional distribution and change management [10].

Chappell [28], emphasize that the definition of Application Life-Cycle Management is not an easy task and that the ALM is of great importance in project management. He states that for this reason it is important to improve the meaning of ALM, which is a

perception, and highlights that Application Life-Cycle Management (ALM) and Software Development Life-Cycle (SDLC) are usually evaluated on equal terms. However, he claims that this simple approach is very limited and ALM is in fact much more than SDLC and supports this with the statement of "*it includes the entire life cycle of an application*". The ALM process is defined as "*to include the process from laying the foundations of the project to withdrawal of the application from the market*".

Microsoft employees need to deliver more work in less time as it is expected by their technology and business managers including Cheef Technology Officers, Cheef Information Officers and IT managers [8]. According to the Forrester research [22] as large as 67% of the software budget has been consumed by the companies for the sake of supporting the current technologies and maintaining the legacy systems. A common lifecycle management solution is required to help balance, examine and communicate with systems that are built to conduct business effectively. Application Lifecycle Management (ALM) offers such solution by focusing on the general harmony and the syncronisation of the business targets and the priorities of IT investments. Optimization of the software development process is based on automation, integration and coordinated approach. It is a common belief of the Microsoft Officials that developing and supporting software has several challenges when it comes to effectively matching with the business goals [8]. The idea is that it is easy to acquire tools to support ALM such as Microsoft Visual Studio Team System (VSTS).

An ALM solution for software development has been introduced by Lacheiner and Ramler [2] in a large industrial manufacturing company. They present the background of the case company and outline an ALM solution (evaluation, piloting, customization, enhancement, roll-out) including the accompanying process management activities. In addition, it describes how the features of the ALM solution were used for process management in requirements management (7 rules), project planning (9 rules), development (10 rules) and product quality (6 rules).

2.3. Tools that Support Application Lifecycle Management and Advances in ALM Technology

Kravchik [35] argues that most important ALM platforms are IBM Jazz and Team Concert, Microsoft Team Foundation Server and Visual Studio Team System and Comverse DiME. Each platform is identified a set of details including breadth of lifecycle support, integration, role-based views, traceability and reporting, platforms support, scalability, security, incremental implementation, integration and interoperability with existing solutions, success stories etc.

ALM contains important building blocks in Sample; Management of claims has methodologies that facilitate the management of such concepts, such as extracting, designing, coding, testing, analyzing living processes, improving the product on the examiner, and removing new versions as a result of the software. ALM also supports the life of an application with tool and process. Tool: Enables people to be more productive so they can produce products faster and earlier. Process: Provides the methods to be followed at the point where the product is of better quality. Software methodologies can be thought of as a small part of ALM. Some of the Software Methodologies are: Waterfall, Scrum, XP (eXtreme Programing), TDD (Test – Driven Development), FDD (Feature-Driven Development), Lean, Agile, Prototype, Incremental, Iterative, V-Model, Spiral, Cleanroom, RAD (Rapid Application Development), DSDM (Dynamic System Development Model), RUP (IBM Rational Unified Model) [36]. ALM can be divided into three separate divisions management, development, operations.

An ALM toolkit has the following submitting solutions can be expected:

- Requires Analysis
- Requests Management
- Modelling
- Design
- Project management

- Change Management
- Configuration Management
- Software Information Management (Framework)
- Load Management
- Software Testing
- Version Management
- Software Installation
- Error / Issue Management
- Measurement, analysis and reporting

The following table shows the comparison of some firms that produce solutions in application life cycle management according to the process tools:

Table 2: The Comparison of Application Life Cycle Management According to the Process Tools

	IBM	Borland	Microsoft
Requires Analysis		Caliber DefineIT : software requirements definition system that enables accurate and complete software requirements definition at the beginning of a new project or enhancement activity [37]	
Requests Management	RequisitePro, DOORS: is a requirements management application for optimizing requirements communication, colloboration and verification throughout your organization and supply chain. [42]	CaliberRM: a requirements management software that facilitates collaboration, impact analysis and communication throughout the software development lifecycle [38]	
Design	Rhapsody, Software Architect, Software Modeler	Together: a visual modeling platform for software architecture design that supports architects, developers, UML designers, business process analysts and data modelers in the accelerated delivery of high-quality software applications	Visual Studio: combining tools with Visual Studio Team Foundation Server, you can apply proven practices to manage your application's lifecycle, from understanding customer needs through code design and implementation to deployment[25]

	IBM	Borland	Microsoft
Project management	Focal Point: provides market - and business-driven product and portfolio management - helping executives and teams to make the right decisions to deliver business, customer and market value. [43]	Tempo	MS Project:
Change Management	ClearQuest: is a bug tracking system that offers flexible defect and change tracking, process automation, reporting and lifecycle traceability for better visibility and control of the development lifecycle. [44]	StarTeam: is a revision control system used in software development [39]	Visual SourceSafe
Modelling	Rose: offers a common modeling tool and language to simplify the work environment and enable faster creation of quality software. [45]	Together	Visio
Configuration Management	ClearCase: is a family of computer software tools that supports software configuration management (SCM) of source code and other software development assets [44]	StarTeam	Visual SourceSafe
Software Testing	TestConductor	Silk Central: enhanced flexibility regarding tool adoption providing improved usability, performance and reporting capabilities.[40]	
Load Management		Gauntlet: is promoting the use of builds as an opportunity for test execution early in the development lifecycle [41]	
Version Management	ClearCase	StarTeam	
Error / Issue Management	ClearQuest		
Measurement, analysis and reporting	Insight		

In a study by McHugh et al. [3], it is argued that as the ALM structure has matured, two focal areas have been formed by leading vendors in this field as the integration of

the ALM segment tools and improvement works of the quality processes. In addition, all suppliers such as Borland, Compuware, IBM, MKS, Serena and Telelogic highlight integration and Serena additionally has supported an Eclipse Application Life-Cycle Management (ALM) to integrate UYY tools around the Eclipse platform. It is also stated that Eclipse is a natural platform for Java development but Microsoft Visual Studio Team System for NET developers provides a platform for ALM along with a number of vendors that provide plug-ins.

In the general, in article [3], it is argued that adopting the right process for the efficient use of the ALM is an important step. It is stated that both Borland and IBM process capabilities exist and Agile may provide Rational Method Composer, which is the new version of IBM Rational Unified Process, in partnership with such service providers as Borland, Exoftware and ThoughtWorks and its own capability maturity model integration consultants. It is claimed that ALM process is the best approach for complex and/or medium and large-scale projects with an appropriate team [3].

Herden et al. [10] claim that ALM technology status includes paper-based, manually-operated, guided procedures to a larger extent. Various recommendations have been made for ALM employees in the companies;

- Manual information gathering by the ALM staff and reduction of repetitions,
- Reducing the amount of required paper -based rules,
- Systematic reduction of the probability of error due to surveillance and misunderstanding,
- Reducing the amount of repetitive, manual management decisions and actions,

might be useful.

Schwaber [22] claims that IT organizations spend billions of dollars annually on the development life-cycle tools, which do not provide them with a proper development and state that for most of the companies, coordination of the application life-cycle management (ALM) –development life-cycle activities remains to be manual to a large extent. While emphasizing that the ALM tools of our day may not provide support

beyond what is achievable with the integration of the ALM from tool to tool, it is argued that future ALM platforms will get better by providing joint service with the tools. The claim that implementation, sustaining and use of these solutions would be easier is supported with the idea that development organizations will ensure that better software is developed. Dr. Schwaber [22] argues that it is very tiring to work without the integration of the life-cycle tools and gives supporting examples stating that These connections increase the efficiency of the ALM and take the ALM to the next level:

1. **Traceability of Interstages Relations:** It is argued that it will associate with life cycle stages such as requirements, models, source code, compilation structures and test scenarios and will help to demonstrate that the software provides the functions required by the work. It is stated that increase in the need to coordinate development as well as the internal and external requirements will make traceability among roles, locations, and organizations a must-have rather than a requirement and it is claimed that traceability for most organizations is a manual process.

2. **Automation of high-level processes.** It is exemplified that in order to control the relationship between functions such as development, analysis and design and the structure and tests, generally paper-based approval processes are used. It is argued that with the ALM the efficiency will be improved by automating these tools and storing all related documents while the automation of the financial services and processes performed will make the work of the developers easier. It is claimed that the process models, which are carried out automatically and electronically on the tools, will be a good documentation for the company. In order to make such documentation reliable, it is recommended to use the ALM system.

Chappell [28] highlights that ALM tools should be of a structure that usually provides functionality to help estimate and plan the projects. Certain tools are designed to manage traditional waterfall projects; some are designed for agile development projects while some others have the power to handle both. The ALM package and most of the ALM tools offer integrated source code management functionality and additionally one of the basic properties that differentiate the ALM tools from project

management tools or problem monitoring systems is the fact that software development process includes QA (Quality Insurance) section.

IKAN ALM tool, which is explained in another article [14], is defined as a meta-model that is capable of integrating, coordinating and managing all different stages of Application Life-Cycle Management (ALM) as requirement definition, design, development, versioning, test phases and finally approval based application development process. Due to this general approach, cost and efficiency of application development process is majorly impacted by the ALM. As a web-based solution and cross-platform IKAN ALM has been presented Application Lifecycle Management. In order to support highly distributed systems and the complex nature of service-oriented architectures, this tool combines the lifecycle management and the DevOps initiatives (continuous constructions and integration). The entire Application Lifecycle Management process aspects are covered by IKAN ALM tool, which provides version and strucure management, manual development, continuous integration, approval process, rebuilding and distribution management. It is claimed that this tool has a full complience with some of the most well-known industry standards including but not limited to ITIL, PRINCE and CMMI. It is stated that while it ensures that your organization implements, controls, supervises, reports and facilitates the best practices and directives for the Application Life-Cycle Management and guarantees compliance to the supervision requirements, it also provides advantages that improve communication with different sides within the IT system.

2.4.Impact of Application Life-Cycle Management on Project Management

Inchki et al. [1] inform that software life-cycle includes such steps as planning, software development, estimations, gathering the requirements, risk and change control methods, business case studies, user interface prototypes and all explanations need to be planned to fully control the project. It is emphasized that with the help of proper planning it will be ensured that necessary checks are made to prevent the project from the aims and the customer has a clear idea about the project and it will facilitate to estimate which outputs will be generated at which stages of the project.

Within the article, it is claimed by Inchki et al. that software projects are long-term and complicated and it is stressed that serious efforts have been made to successfully

end the project, CHAOS Report, published by Standish Group for 2009 and given as a reference, was reviewed and it is explained that 68% of the latest software projects have been finalized unsuccessfully [1]. They listed the reasons for the failure of the projects as follows.

These are:

- Lack of user input
- Incomplete requirements and specifications
- Changing requirements and specifications
- Lack of management support
- Technological insufficiency
- Lack of resources
- Unrealistic expectations
- Unclear goals
- Non-realistic timing estimates
- New Technology
- Other factors

By utilizing these statistics, they explained in detail the opportunity areas to occur to make these projects successful as well as displaying the failure factors of the software projects. It is argued that methods and processes to be followed carefully will ensure that these projects are completed successfully.

As it can be seen, one of the most important factors that the data shows is that the customer needs cannot be gathered and analyzed correctly and it is argued that it will be inevitable for the project to fail due to the mismanagement of the requirements, which are the most fundamental factors for the proper implementation and success of any project, regardless of the technical skills. It is emphasized that it is not an easy task to gather the customer requirements, define the customer needs and then try to produce a solution that includes these features, in addition, after having fully understood all details related to the customer's business, deciding how and by which methods these will be developed, submitting these decisions to the customer and receiving the approval and therefore narrowing down the necessary properties set will help to achieve the most ideal conditions in terms of time and budget estimations. It is stressed

that this is the stage, where the business analysis and customer relations roles are under the spotlight.

It is highlighted that the stages of preparing the project plan with the start of the project, documenting the needs of the requests analysis team and recording these in the form of a software requirements specification document are important. Also, the user interface prototype models (mockup) that describe how the properties desired by the customer are presented in the resulting application are clearly stated to be an important part of the requirements analysis phase.

Careful monitoring of requirements changes, effective sharing of these both with the client and the development team, will not only have a positive effect on the success of the project but also these software process stages need to be supported with the new generation ALM tools to encourage interoperability and provide application life-cycle integration. With the transformation of software development, more and more tools have been introduced for this purpose over time and the idea that software development is supported by special-purpose tools and that later on there are simple tool sets that do not have very strict integration is supported. Upon the widespread acceptance of the concept of ALM concept lately, modern comprehensive life-cycle tools to serve this purpose are presented as a solution. However, since they are the products of certain firms, it is claimed that they cannot overcome the platform constraints and will prevent an effective and sustainable work and also it is stated that many companies try to market the tool set, which is made up of inter-corporate tools developed for each process, as an ALM solution.

For each one of the tools used in the application life-cycle, they argue that they present different images of the project and that all data to be used/produced throughout the project should be organized in order to integrate these tools and ensure their communication. In addition, all tools will be designed around the common control data, the integration of tools therefore integrated monitoring of the process will be achieved. Additionally, problem and error management, load management, modeling and simulation, requirements management and installation management are specified among the building blocks of the ALM.

An ALM tool set is expected to provide the solutions below:

- Requirements Analysis

- Requirements Management
- Modeling
- Design
- Project Management
- Change Management
- Configuration Management
- Software Information Management
- Software Testing
- Version Management
- Software Installation
- Error/Problem Management
- Measurement, analysis and reporting

Thanks to the measurements (i.e requirements, software development, verification and validation and other processes) to be carried out on the ALM system, the project manager will carry out the management activities by using the current vital data of the project and it is also argued that the periodical reports to be presented to the customer will be produced through this system and this will lead to more effective running of the project review meetings. In this way the system will not only integrate data flow but also manage the control flow. As a result, integrated management of the project stage is the aim.

According to the claim in the article published by the Microsoft [8], unless managed well, software projects are faced with the risk of failure. They can easily deviate from the aim, fall behind the programme, exceed the budget, and fail to deliver the sufficient quality levels or provide a concrete business advantage.

It is argued that lack of communication and misunderstanding between the business sponsors, may contribute to the errors at most of the times. Four predominant reasons of project failure, which are identified by the Microsoft officials, are as follows:

- **Project Status visibility issues:** This is one of the main Project Management issues, which may include lack of responsibility, accountability, signiture and control points. Lack of stakeholder engagement, inability to make accurate

estimates and failure to adjust the project programme accordingly are indicative of project management shortcomings.

- **Ineffective team communication**: Coordination efforts between functional, geographical and organizational boundaries are a particularly difficult communication problem.
- **Balancing business demands with project risk**: The combination of the problems such as unstable and poorly defined requirements, changing scope, unclearly defined estimates, uncertain business targets and rapid evoluation in technology increase the risk.
- **Unpredictable work completion times and the quality of the final product:** Establishing the ballance between service requirements, functional requirements, quality and budget is very challenging.

Pícha and Brada [11] stress that Project Management and Software Process Improvement (SPI) are significant parts of software engineering and state that there are many project management techniques and tools such as Application Life-Cycle Management (ALM) and a number of software methodologies have been defined. Pichler et al., in this article [11], stress that software engineering made significant advances in terms of providing better tools and guidance for the software development process in the recent years and allege that Application Life-Cycle Management (ALM) includes many methodologies, best practices and supporting tools. These tools later are included in a certain software process and are used to provide data within the scope of Project Management (PM).

Chappell [28], exemplifies the ALM foundation as a life-cycle of an application similar to the life-cycle of an individual and it is claimed that "*Before Agile Manifesto, the previous waterfall approach in software development was completely inadequate*".

In this context, it is highlighted that the project is prone to cost overruns and extended deadlines [10]. With these problems, the project managers understand the importance of how efficient they can be with process management. The ALM is basically defined as combining disciplines related to all aspects of the software delivery process. Instead of a series of business analysts working to define the vision and requirements, it is stated that the concept of ALM, which provides regular stages such as delivering design to developers, handing the code to test users, delivering the product to the IT

support team and distributing the product, has started to be integrated into the company projects. It is highlighted that ALM tools should be adaptable to methodology and procedures.

2.5.The Correlation Between the Application Life Cycle Management and Business Process and Process Management

In an article published in 2011 by Hermann Lacheiner and Rudolf Ramler [2], it is noted that Application Life-Cycle Management (ALM) has been widely supported in recent years and that ALM solutions attract the attention of many software developer companies. That being the case, it has been decided to introduce an ALM solution for Software Development in a large industrial firm. Process improvement methods are addressed in the article. The ALM solution provides that the project life-cycle is completed using team evaluation, introduction and process improvement activities. Within the scope of the article, this article investigates in what extent does the provided features of ALM for software development contribute the effective management of the software processes. Implementation methodologies of ALM solution for process development, process implementation, process documentation and process monitoring were illustrated in this paper.

Forrester's argument about the ALM structure is that the development lifecycle activity implementations and the processes covering such activities, including the input and output correlation management of these activities, will facilitate the reporting procedures as a whole.

Therefore, it is stated that "*Although the ALM tools are considered to be only a product market, they are more than that and ALM is a product category as well as a discipline*". Some of the key points of ALM processes such as analysis, traceabilty and reporting can be done automatically rather than manually by tool integration, which can make the entire process more efficient. It is emphasized that ALM does not support certain life-cycle activities and has a structure that keeps them all in sync. It is stated that the ALM provides the coordination of the activities carried out by the analysts, system architects, developers and testers. ALM Solution was defined by Herman Lachene and Rudolf Rams [2] as the integration of Lifecycle tools. As it is understood from this

point of view, it is not the tools that constitute the solution of the ALM but the links. It is stated that integration of the life-cycle tools lies in the foundation of the ALM concept.

- Traceability of relations between stages

Associating such life-cycle stages as requirements, models, source code, assembly structures and test scenarios, helps to show that software work gives the desired functions. Increasing need to coordinate development as well as internal and external compliance requirements renders traceability among roles, locations, and organizations a must-have rather than a requirement.

- High-level Process Automations:

Organisations that are specialised in software development widely use manual approval methods in order to control the relationship between certain functions such as Analysis & Design and Structures & Tests. This paper-based operation and related documentation storage can be done automatically, therefore more efficiently by adoption of ALM.

- Tracing the progress of development:

Following-up the progress of software development projects is one of the major issues of the managers as they have limited visibility into this progress. The data that is used for this is collected from subjective references instead of objective references. One of the customer comments clearly states this: "*We are reporting in the same way as we did for about 40 years. Everything is progressing manually. Weekly status meetings and progress reports are works that require continuous follow-up. Instead of providing follow-up to ask whether people are doing their jobs, it will be more efficient for our company to provide automatic follow-up of the works held in a place.*" With this statement, it is observed that business-task follow-up systems provided by the ALM have become a need for the companies.

McHugh et al. [3], state that advances are observed in the teams for open source software projects, agile (Agile development movement and especially ALM tools). They supported their claim with the reports on software projects published by the Standish Group CHAOS for years and they state that they are frequently used in large-scale projects in particular for software development to support an uncertain situation.

It is noted that the ALM includes such project commencement and work statuses as production performance management, requirement definition and specification, architectural design, detailed design, construction, test, building and distribution and three disciplines operate in each stage throughout this life-cycle. These are project (portfolio) management, requirements management and change configuration management (CCM). Information on these issues is provided.

2.6.Impact of Application Life-Cycle Management on Competition and Development and its Importance for the Companies

Chappell [29] argues that each corporate company should pursue strategies to separate itself from its competitors. For the vast majority of these efforts, he claims that special applications are necessary to achieve this differentiation. He argues that creation and operation of these special applications will be realized by the application life-cycle management (ALM) process. He states that the link between ALM and its business strategy is not always clearly understood, however, he highlights that from the right perspective, a modern organization should be good at strategy and thus good at ALM to ensure long-term profitability.

Chappell [29] emphasizes the essence of his business strategy. He shows as a possibility that a firm may do different things from its competitors, for example may offer different products or attract different markets. Alternatively, he also highlights that the same things can be done in different such as providing a lower cost service. In both cases, it is the ability to differentiate an organization from other firms, which characterizes an action as strategic, and claims that it will allow offering a unique value to the customers. Claiming that the first company to implement a successful innovation in the sector gains a significant competitive advantage Chappell defends that the second firm gains a fair amount of advantage, too.

In addition, Chappell [29] argues that being good at ALM is an important part of gaining a competitive advantage and that it should be a basic goal to dominate ALM in every organization, where special software is important, namely almost in all organizations in our day. In another article, Chappell [30] argues that business

processes have become more dependent on software and then claims that it is more important than ever to have the right software. Defining the set of activities required to create and run custom applications as the application life-cycle management (ALM), Chappell stresses that ALM supports a number of business processes and includes a critical business process on its own. He highlights that any company that develops custom software should take the ALM process as seriously as possible in other important business processes. He argues that in custom software development, being better than other institutions could provide a significant competitive advantage.

Stating that organizations are defined by business processes, Chappell [30] emphasizes the need to understand their processes to understand what an organization is doing. In the real sense, he suggests that improving an organization –making it more responsive to change, being more profitable and valuable– has the same meaning as improving business processes. This is why the ALM process is so important. Stressing that in fact, any corporate company that develops custom enterprise software should take the ALM process as seriously as other business processes, Chappell claims that ALM might be even more important.

Chappell, who stresses that being better at developing and using special software than the competitors could provide a significant competitive advantage, similarly argues that "*being worse may become a significant disadvantage for you*". He once again highlights the importance of the ALM with the statement "*Unless your organization considers ALM as one of the most important business processes, it is time to change this view*".

2.7. Secure Software Development Within the Scope of Application Life-Cycle Management and its Standards

Beydağlı et al. [5] examine secure software development models, standards and mainframes from past to present that have emerged in the development of secure software, which has gained importance with the widespread use of business applications in computer environment and increased internet usage. The models being already used to develop secure software are compared with the security approach presented by the Common Criteria (ISO 15408) and the impact of Common Criteria standard on producing secure software is discussed.

Beydağlı et al. [5] claim that with the first emergence of the computers software development process also started and this process goes back to 1040s. The most important deficiency in the software developed in the early years was the inability to complete the software projects on time and the lack of quality (documentation, functionality, excess labor). In addition, the rapid change of technology has caused a number of software to be disabled due to lack of speed and functionality and to re-project work. In 1980s, Fred. P. Brooks states in the article "*No Silver Bullet*" that it is possible to succeed in this issue with one parameter. After identifying the problems, new languages (C ++, Java etc.), new tools, new methodologies and new disciplines were tried to be developed. With the widespread use of the Internet, the software started to be abused by malicious or unconscious users due to the security vulnerabilities/gaps and the secure provision of the works in our daily life such as e-commerce, e-government and e-health through software has become risky. First studies on software/hardware and system security assessment, started with the publishing of the TCSEC (Trusted Computer System Evaluation Criteria) standard, which is also known as the Orange Book, by the US Department of Defense in 1983.

In Europe in the 1980s; Britain, Germany, France and the Netherlands established their own security test methodologies. These countries then set up the ITSEC (Information Technology Security Evaluation Criteria) standard in 1991 to eliminate the differences between the assessment standards and to ensure that assessment made is valid everywhere. Canada, using ITSEC and the TCSEC, published the national CTCPEC (Canadian Trusted Computer Product Evaluation Criteria) in 1993.

Within the scope of the article, process improvement and evaluation in CMMI is examined in four categories. These are Project Management, Business Management, Engineering and Support. It is observed that CMMI processes are not directly related to security.

Federal Aviation Management Integrated Capability Maturity Model (FAA-iCMM Federal Aviation Integrated Maturity Model) FAA-iCMM is widely used in federal aviation management. FAA ICMM model claims that it will provide a model that is made up of best practices aiming to make advances in the large software systems (enterprises), which also include outsourcing and source management. The latest

version includes integrated large software management, information management usage / transition / disabling and transaction / support processes.

Secure CMM/ Secure Software Methodology

(Trusted CMM/Trusted Software Methodology (T-CMM TSM)) Strategic Defense Initiatives (SDI) developed a new model called as "*Secure Software Development Methodology*" at the beginning of the 90s. Then this model was called as secure software methodology (Trusted Software Methodology (TSM)). This model is composed of levels provide resistance against unconscious developers' errors at the low levels and against malicious software attacks at the high levels. TSM is then combined with CMM (Harmonized) and Trusted CMM emerged [4]. Although TCMM/TSM is not widely used in our day, it can be used as a resource in software development process in the future.

System Security Engineering Capability Maturity Model (SSE-CMM): SSE-CMM is explained as a process model that can be utilised for evaluating and developing the security engineering capacity of an organisation. SSE-CMM developed an acceptable framework for evaluating security engineering practices according to generally accepted engineering principles. Such a framework provides performance measurement and improvement in the application of safety engineering principles. The SSE-CMM was published as an ISO / IEC 21827 standard and is currently available as version 3.

Clarifying the software used in information systems, it is determined that there are continuous gaps in the software due to not being developed as secure. These gaps and weaknesses can be consciously or unconsciously misused by users. Users do not want such gaps in the software or systems to be used in the information systems. The most effective way of reducing these gaps and their impacts is the secure software development. Important secure software development standards, mainframes or models, which are already used in secure software development, are addressed in this article. As a result of the review, none of them meets all the criteria for developing secure software. It is estimated that Common Criteria standard, main objective of which is the evaluation of the security functions of IT products, can be a secure development guide for secure software developers, especially at high security levels.

The fact that the Common Criteria standard has been directly adopted by 25 countries and has an evaluation methodology and has been published as an international standard, puts it at the forefront of security compared to other systems. In particular, it includes more security measures in respect of physical and logical security of the development environment compared to other standards and mainframes.

2.8.Importance of Application Life-Cycle Management in Software Engineering and Software Life-Cycle

Klespitz et al. [6] highlight that development lifecycle management is an important aspect of software development. Also, software development effectiveness and the end product maturity is defined by development lifecycle management. They argue that in the critical development, including the medical device delopment, it is highly important to implement an appropriate management system. Such developments needed to be well documented in order to prove secure operation of the devices in question. As a result of this study a case study has been prepared which assists to the company to develop a life-cycle management system by identifying the relevant requirements, especially the software develpment aspects of medical devices. It is argued that the answers to these questions will provide an environment for optimal selection by comparing the quantities among the prospective management systems or emphasising its negaative justifications.

According to Klespitz et al. [6], one of the most significant concerns of the companies is the quality of software. Hence, more quality standards should be observed for the medical devices. These standards are identified as "ISO 13485 Quality Management Standard, ISO/IEC 14971: 2007 Risk Management Standard or ISO/IEC 12207 Standard for Software and ISO/IEC 15288 Standard for Systems". It is argued that the development process should be well documented to meet the requirements of these standards. An appropriate ALM system establishment could reduce the documentation effort significantly. Although this does not require standards and rules, plus some other studies are available like Agile development approach, this is not quite widespread. Therefore, the companies that adopt plan oriented method for software development are mostly targeted by the ALM Systems. The number of available Application Lifecycle Management systems in the market have been increased, therefore it is

suggested that the appropriate ALM system should be selected which shall meet the needs of the developmet team. As well as the developers needs, the created ALM system should also meet the needs of the managers at the same time.

Maksimenkova et al. [12] argue that main professional lifecycle processes like analysis, design, development and maintenance (see SWEBoK, ITIL etc.) are simulated by Software Engineering (SE) processes. Supported Collaborative Learning (SCL) techniques should be driven by the professional tools in order to meet both industry requirements and educational needs. Maksimenkova et al. in their article [12] aim the following for adaptation of the professional software:

- Simplify information security requirements and rules,
- Optimize processes and constraints for small groups and multi-phase works,
- Integration of API and ALM software,

When these issues are applied, it is argued that the trainings provided within the company will be beneficial to the ALM structure in terms of habit.

Gross and McInni [16] examined the SDLC concept known as the software development lifecycle or system development lifecycle, and some software development methods, which are referred frequently in literature depending on this concept and can be considered as fundamental. These software development models are waterfall model, fountain model, v-shape model, iterative model and spiral model. In addition to introducing the basic features of the models, the examination of the superior and weak sides of the models and their position within the system lifecycle are also examined. In [16] they also state that Software Development Lifecycle (SDLC) concepts is a cycle approach and SDLC stages are as follows:

- Planning
- Definition
- Design
- Development
- Integration and test
- Application

It is claimed that project management models usually start with waterfall model, the waterfall model is much more static, for years there has been no need for any other requirement, but it is suggested in the article that this situation is slightly different in software projects. Statement by a project manager "*Sometimes the requests until the redesign of the project may be received successively on the same day*", highlights that waterfall model is not a suitable model for these projects.

When the method development is reviewed on the level of idea, the study referred most in the literature without using the name "*waterfall*" belongs to Royce [51]. The remarkable aspect of this article is that it uses the waterfall model approach without using the name while analyzing the development processes and in general describing the mistakes and problems in the software world. The original method proposed by Royce is as follows [51]:

- Requirement Specifications, product requirement document is published at the end of this stage
- Design, software structure is developed at the end of this stage
- Construction, coding and implementation stage and software is developed in the end
- Integration
- Test and debugging
- Installation
- Maintenance

It is possible to talk about other models developed in addition to the original waterfall model proposed by Royce but in the original model the transitions between stages are clearly defined and designed without return. The most important difference of the model from other varieties is that returns between the steps are more successfully defined.

In the article [17], Liao et al. [17] highlight the increasing technical complexity of the systems, the need tendencies for reproducible and predictable process methodologies, guidance of the system developers to create system development models. Organizations increase the need for the automation of various activities with their expanding operations thus creating industry-based standard and structural procedures

or methodologies to facilitate the transition from the manual to the automatic system. The concept of system lifecycle models highlights the need to follow some structural approaches to create a new or enhanced system. For this, multiple models have been proposed including prototype, V shape, waterfall, rapid application development etc. Various Software Development Lifecycle Models have been compared and analysed in this paper.

Liao et al. [17] provide a better understanding of the concept by defining that a software development process, also known as software development lifecycle (SDLC), is an applied structure for software product development. Considering that they are generally considered a subset of the system development lifecycle, several models have been defined for these processes, each describing the approaches to various activities taking place in the process. The method of selection, implementation and moditoring the lifecycle for software is defined by the ISO 12207 standard. The processes of software development are separated into multiple logical phases which allows software development companies to develop an efficient software product within a predefined schedule and budget. Liao and other article authors reported the comparative results of software development lifecycles, where software lifecycle models are focused on detailed software development lifecycles. They have used a conceptual research approach and a meta model for software process lifecycle in order to enable a reasonable explanation. Software development model comparisons and the basic principles are presented by Jovanovich D. et al. In the context of the article, problem solving is composed of the steps as follows:

1. The problem comprehension of Compatative SDLC Model Analysis
2. Decision of a solution plan
3. Encoding the solition that was planned
4. Real program testing

In order to perform these steps efficiently and accurately, it is emphasized that some metodologies and procedures are needed as each step can be extremely complicated. Apat from the complexity, some of the main activities can be quite large that they can only be addressed in multiple steps. For instance, a large sotware system design is claimed to be divided into a number of different design stages, commencing with a

high level design which specifies the system components based on a design that is in more detail. The main software system development stages or activities are:

1. System Requirements Determination
2. System Design
3. Coding (Software Development)
4. System Testing

In addition to the activities carried out during software development, some activities are carried out after the main development is completed. It is said that it is an installation stage related to actual uploading of the system to the computer systems of the client and then testing. Maintenance is defined as a step that starts after the software has been developed and it is argued that some components of the software should be protected as there are certain errors in the system that need to be removed when identified not because certain components of the software are "*worn-out*" and need to be changed. Therefore, it is emphasized that maintenance is inevitable for software systems.

Taya and Gupta [19] exemplify the sampling of multiple methodologies for software development life cycle. These methodologies vary according to the area, requirements and process of software projects. In this context, all methodology types are examined and explained in detail. Within the scope of the articles examined in this section, the SDLC concept known as software development lifecycle or system development lifecycle and some software development methods, which are frequently mentioned in the literature in relation to this concept and can be considered as basic, are examined. This software development models are waterfall, fountain model, v-shaped model, iterative model and a spiral model.

Software development methods and software development methodologies of ALM tools and their operating logic and the ease they offer are explained in detail.

Taya et al. [19] argue that in the planning phase, the methodology of the software could be identified and be any of the examples of Waterfall, Scrum within the scope of Agile, Kanban. Depending on the methodology chosen, the ALM tools constitute a process monitoring system for the software process [27]. When the comparison table

for IBM JAZZ, Atlassian JIRA and Microsoft Team Foundation Server detailed in Table 2 is reviewed, it is seen that ready-tables are created for Kanban. Thanks to this feature, it is easily identified by the Project Members the stage, where the project is, by reviewing the "*To Do*", "*In Progress*" and "*Done*" tables [14,15,16,17,18,19]. It is also argued that there are suitable platforms in the ALM tools in terms of determining and planning Scrum meetings.

Simao [21] stresses that the importance of software in the world is constantly increasing so is the size and development impacts its importance increases. He states that these effects have a great importance on the way the software is developed. In the software development process of large projects, large amounts of money and time are required to be used correctly to present the product at minimum cost. With the evolution of software development, different approaches are provided to reduce cost and time without conceding on the product quality. The benefits and criticisms of each approach have been addressed, but it is revealed that all of them have the same objectives as offering the best quality products in the fastest and cheapest way. When we search for a Software Development Methodology (SDM), very different approaches are selected for many different perspectives and even for the same perspectives. These SDMs determine the ways, in which teams and activities are organized, and they are claimed to be a good guide to achieve the goals. Therefore, a big question that arises when starting a project is claimed to be the methodology to be chosen. Most companies adopt one or more methodologies and then continue the project with a specific one while it is stated that in times of economic crisis, where budget and resources are more limited than ever, a bad choice or abuse of a methodology may cause more money and resource expenditure than expected.

In addition, Simao [21] elaborates the methods of software development. Waterfall is still one of the most widely used methodologies in the world (or methodologies based on this model), so it is argued that it is an important methodology that can be considered in the analysis. It is informed that the requirements are identified from the beginning and are not changed during the process and it is identified that there are bot advantages and disadvantages. In engineering projects, it is stated that the rate of choice is low due to changeable area and efficient process evaluation or adaptation problem and the design process continues with the documents rather than practical solutions that enable a more efficient design in defining different design levels.

General features:

- Activities at every step of the waterfall are fully fulfilled. This is a condition for moving to the next step.
- A document is created at the end of each phase. For this reason, the waterfall model is document-guided.
- The software process is linear, meaning that the activities in the previous phase must be completed to proceed to the next phase.
- User involvement is possible during the initial phase. User requirements are identified and detailed at this stage. There is no dialogue with customers and users in the subsequent design and coding phases.

Problems:

- As one cannot move to the next phase without completing the previous one, all requirements of the customer need to be defined in the first stage. As the vast majority of software developers experience;
- At the beginning of the project, customers are not 100% sure of exactly what they want
- Customers may find it difficult to express their wishes clearly if they are not involved in software technologies.
- Details of the customer's actual requests are not noticeable at the beginning of the project; they are brought to the surface in the process.
- As the customers see their wishes come true, they tend to make changes as their project begins to produce visible results.
- External conditions are in a change and it is inevitable for the projects to be affected by them. With the waterfall method, the software system the customer wants is completed at the end of the project. Only at this stage, the customer will be able to test the software system. The customer has to accept and use the completed software system with all its pros and cons. Therefore, the principle that all requirements are defined and there are no second thoughts means to ignore any changes that may occur in the product development process.
- Understanding the customer request and creating details documents for all requirements, software developer to read and implement these, to redo everything

in case of any error likely to occur during the test phase, will make the realization of the project longer.

- Detection of the errors made at the beginning may take a very long time. Debugging will increase the cost.

Agile Model is released to solve the problems encountered in the software development process (exemplified by waterfall models) and developed based on repetitive software development model and promotes change and software delivery in parts and frequently. Agile Model aims to divide the work into parts rather than moving to design-development-testing stages after determining all the requirements at the beginning as in the Waterfall model and completing the analysis phase and run this process.

Within the scope of the article [21]; two different software development methods detailed above are preferred by the project team and can be used appropriately.

Chappell [31], in a traditional development process, made some statements to claim that the uncertainty of the requirements is not present. In the classical waterfall approach, for example, an organization creates detailed plans and precise schedules before writing any code. However, he argues that these plans and programs are rarely complied with and that the main reason for this is that, although we use the term "*software engineering*", code writing is not similar to other types of engineering. In traditional engineering projects –when constructing a bridge or factory– it is defined as "*it is generally possible to define the consistent requirements in the fore*" but in agile software development it is stated that creating consistent requirements is generally impossible for the customer because they may not know what people want at all times. Chappell [31], emphasizes that event waterfall and agile methods have different usage areas in our day, they are the most leading software development processes in the area of software. Different from traditional development processes, an agile process does not try to determine three variables at the beginning of the project (time, budget and work scope) for sure.

2.9.Impact of Application Life-Cycle Management on Communication Management

Biró [7] highlights that communication is very important during software projects and that it is required to determine the requirements of all departments and follow a course that complies with that. Within the scope of this article, the benefits of the ALM Tools are mentioned for the continuous monitoring of the works. Biró [7], argues that the most important reason of project failure is requirements and specifications concepts and highlights that what customer really wants and expects should be fully understood and included in the project. It is emphasized that information should be transferred in a continuous and efficient manner to the ALM structure according to the role definition and the project stakeholders should be informed of these changes in the project in a timely and at the required rate thanks to this information flow. Thus, he advocates the effective implementation of change management. For example, a change to be made in a requirement, which is still being developed, will be informed to the developer over the ALM instantly and efficient requirement management and change control will be achieved.

Kääriäinen [13] argues that Project Management and Software Process Improvement (SPI) are important components of software engineering, while many project management techniques and tools are available such as Application Life-Cycle Management (ALM) and many software methodologies, process metadata models, and best application descriptions, which is the main theme of the article. While Kääriäinen suggests that the tools play a major role in the implementation of the processes, he stresses the importance of ALMs in keeping track of the tasks and keeping the communication between the departments under control in Software Projects.

Kääriäinen in his article, stresses the importance of communication between the departments stating that "*It is an indispensable factor in order for the departments to operate in an integrated manner in the process software lifecycle*".

JunWu and JunLing [20] claims that in the process of software projects, the customer concept is very important for the success of the project. It is stated that it is not always possible to produce a product as the customer wants, and in this regard, customer satisfaction and strict communication with the customer will affect the success of the

project to a large extent. It is argued that Customer Relationship Management (CRM) is not only a technology but an organization's philosophy of dealing with its customers is a comprehensive, customer-oriented approach. These policies and processes include preliminary customer service, employee training, marketing, systems and information management. In terms of the objectives of a CRM strategy, they highlight that a company's special situation and the needs and expectations of its customers should be taken into consideration. In view of this problem, it is detailed with the processes that the ALM tools maintains the customer-company relationship.

JunWu and JunLing [20] highlight that satisfaction, which occurs during the software by means of constant communication, of the customer, who acts as actor/stakeholder throughout the project period, is very important for corporate firms. Customer relationship management (CRM) is basically defined as the function of establishing a long-term relationship with a customer and it is claimed that protecting an existing customer is less costly than finding a new customer. Customer relationship management is defined as consisting the processes used to track and organize relationships with its current and potential customers and it is stated that CRM software is used to support these processes. It is alleged that marketing strategy can be supported expanding knowledge in such issues as identifying the customer profile of the organization, keeping the customer, improving product proposals (by better understanding the customer needs) and identifying the most profitable customers of the organization. It is stated that ALM tool systems are being used to support these processes.

2.10. Application Life-Cycle Management and Traceability Evaluation

As it reported in an article published by Microsoft [8], when the tool integrations are not done well enough it becomes quite difficult to provide automation, tracebility and accurate progress tracking. Application Lifecycle Management is a very significant matter for project and configuration management systems, modeling and visualisation tools, integrated development environments, problem monitoring systems, compliers and source code managers. One of the many crucial features of ALM is that it allows sharing a common up-to-date information pool between the project stakeholders

including project managers, sponsors, system architects, developers, testers, system admins and the end users. Typical activities including requirement collection, solution modeling, desing, coding, testing, distribution and problem monitoring are supported by the ALM. The outcomes of these activities are connected by the ALM tools.

According to the Microsoft officials [8], while introduction of the ALM strategy would take a considerable amount of time, companies should focus not only on the tools but also on the practices and improvement efforts.

In an article by Giuliano Antoniol and Michael_Vierhauser [15], it is suggested that software and system engineers will see the importance of traceability to develop higher quality products at increased levels of complexity and scale and that integration and approval management, problem monitoring and automation at each step during the ALM process will further shorten and facilitate these processes.

2.11. Applicatıon Life-Cycle Management and its Contribution to Quality Management

Tüzün et al. [9] stress that accurate and complete reporting is very important for the progress of the process in software projects. These reports can be literature survey, requirement analysis, design document, compliance with quality standards, test cases and test plan. When the requirement analysis report is prepared, it is argued that these requirements should be supported by the software side as well as various quality standards and security standards. Identification of the quality standards and security standards requires detailed reporting. Application Life-Cycle Management tools provide a powerful reporting system that will help software projects succeed. The articles in this section include the importance of documentation in software projects and the advantages of the ALM tools for reporting.

Tüzün et al. [9] highlight that it is very important to report all processes during the project. In the context of the article, it is stated that the projects have many functional/non-functional requirements and they should be in compliance with the required standards. The importance of security rules, security standards and quality standards is emphasized especially in critical software such as medical device software and defense industry. At present, although reporting systems continue automatically, some companies are still in paper –based and progress manually, which is reported to

be/is difficult both for report follow-up and the user. It is suggested that the ALM is highly developed in this respect and that it offers a strong reporting and project management follow-up system.

It is argued that the ALM solution is provides a benefit, which is not only effective for software development but also for process management. It is emphasized that process management is very important in this life cycle process, which starts with correct planning and reaches the end-user. The process starting with requirement gathering and delivery to the end-user is the software life cycle.

In a publication by the International Electrotechnical Commission [26], standardization provides processes that support the definition, control and improvement of life processes used in an organization or project. In the paper published by the International Commission [26] it is stressed that standardization provides processes that support the definition, control and improvement of the lifecycle processes used at the organization or within the project.

It is especially highlighted by the International Commission [26] that institutions or organizations can benefit from these processes when they are building their systems or supplying their projects. In this way, it is claimed that the processes are endured to be managed in a healthier way and that in parallel, error-free and quality products are produced. In addition, it is stated that wasting time and labor force will be prevented.

2.12. Evaluation for Application Life-Cycle And Product Life-Cycle Management

Ötleş et al. [18] argue that enterprises of our day face various challenges arising from continuous innovations, world-wide cooperation and complex risk management. In the value chain, it is stated that intellectual gains, which are in the form of product and process data, must be accessible to all and if these problems are addressed, PLM (Product Life Cycle Management) is proposed as a business understanding combining people, processes, business systems and the entire lifecycle of the product in recent years. Product development is defined as a complex step which requires interdisciplinary cooperation and is based on team work. PLM is stated as a complex process dependent on the inputs by many individuals, groups, organizations and even

the society collaborating for integrated product design and development in the competitive, demanding and economically challenging world of our day.

Benefits of the PLM according to Ötleş et al. [18]:

1) Help to distribute more innovative products and services in a shorter time,

2) Shorten the time to the market,

3) Provide more comprehensive and cooperative relations with customers, suppliers and business partners,

4) Strengthen communication between departments,

5) New developed products will increase the success rate,

6) Efficiency will reduce the cost,

It is claimed that in a process where the product is developed without PLM, firstly small errors made in each area will become a big problem in the development of the product and that this product will extend the time to reach the market. There will be a lack of communication between different disciplines and employees, who are not in cooperation, and ultimately it is argued that productivity, efficiency and product quality will decrease.

In the publication by the International Electrotechnical Commission [24], it is argued that ISO 14971 Risk Management Training in Medical Devices is required for all medical device manufacturers. It is claimed that identifying the risks in a product's life cycle in medical devices, evaluation and checking helps you implement a risk management framework within a continuous compliance and the claim that you will have all tools to allow managing your risks in compliance with an international standard at the end of the training is supported with the necessary explanations.

Compliance to this standard is alleged to provide the following benefits;

- Ensure that your medical devices comply with ISO 13485 Medical Devices Quality Management System,
- Ensure the continuity of this compliance with the product lifecycle compliance,
- Provide competitive advantage in ISO 13485 accreditation,
- Increase customer satisfaction and market access,
- Encourage professional development and build risk awareness

Frakes et al. [27] report that contemporary businesses have to face a variety of challenges arising from continuous innovations, worldwide collaboration and complex risk management. On top of these problems, PLM (Product Life Cycle Management) is proposed as a business understanding combining people, processes, business systems and the entire lifecycle of the product in recent years. It is emphasized that relation of the ALM tools with the product lifecycle management is important in this regard and it is argued that "*receiving the project, implementing this project within a correct process and reaching a correct product is an expected result*" at the corporate companies. In addition, Frakes et al. [27] argue that a product lifecycle management cycle of a healthy software process increases product quality and productivity. Product life cycle management will help to distribute more innovative products and services in a shorter time period, to shorten time to the market, to establish more comprehensive and collaborative relations with the customers, suppliers and business partners, to strengthen communication between the departments, to increase the success rate of the newly developed products and decrease the cost due to efficiency [1,9,24,25,26,27]. For this, application lifecycle management and product lifecycle management should operate in an integrated manner. It is emphasized that this will facilitate the follow up of product life cycle process both in document and process management by means of ALM tool.

2.13. Summary

To sum up, the ALM process adoption is becoming an uprising trend among the software development organisations, hence, a number of ALM tools have been proposed [23]. Although recent developments in ALM have heightened the need for more attention from software engineering researchers, there is little published data on ALM [17,18]. This investigation will enhance our understanding of the industrial adoption of ALM practices in a large-sized software company. Therefore, this study makes a major contribution to research on ALM by demonstrating a rigorous empirical assessment for exploring the implications of ALM tools and practices.

It is emphasized that Application Life-Cycle Management includes all processes related to the development, growth of the application, its customization in line with the feedback from the customer and its maintenance for a longer lifetime. For

these matters, there is a need for a continuous process and the ALM is a concept that fulfills this requirement and the primary objective of the ALM structure is to increase the software quality and reduce the costs [21, 23].

To address the current needs (e.g. improve productivity, decrease maintenance costs) for managing a set of delivered products, action research will be conducted by researcher. This thesis will present the results of an action research project aimed at exploring [2] the possible benefits of implementing ALM in a large-sized software company and [35] the outputs of the assessment to investigate the potential constrains, and ultimately quantifying the experience that is gained during the transformation process.

CHAPTER 3

3.METHODOLOGY

3.1. Introduction

This section explains the methodology behind the study. It starts with the descriptions of qualitative, quantitative, mixed and action research methodologies. After then, it explains the reasons of selecting action research methodology and attributes of the study.

3.2. Qualitative Research

According to Crescentini and Mainardi [47], during the implementation of qualitative methods, many stages can be monitored in turn: determining the general research questions, collecting information, determining the transformation in the data, interpreting the data, completing the conceptual and theoretical studies, writing/conclusions. It is argued that monitoring these steps is of great importance in clarifying the research. At the same time, selecting the content and the form of writing will strengthen the reader's understanding of the process. However, considering that the qualitative research process is often much more iterative, it is argued that all content should be well defined at the outset.

It is emphasized in the article that how and why something has become a research question should be questioned in the beginning and it is argued that information on different research traditions, the owner of the research questions and how it has been developed should also be included. In this respect, it is argued that it can be observed that research results are developed as a result of the interaction between the researcher's goals (individual, ethics, etc.) and the researcher's theoretical frameworks. The researcher's theoretical frameworks include all previous research, findings, or theories on the issues that the researcher has triggered.

In terms of addressing the steps, while determining the General research questions, which is the first step, it is important to specify clearly why this research question is determined and why it should be investigated. In the second step of the Information

Gathering section, the resources should be clearly identified and the related person or persons should be contacted. Data analysis should be done. In determining the transformation in the data, which is the third step, a conceptualization process that includes the "*things*" found by the researchers, theory and aims starts and they are turned into knowledge. The researcher should clearly state this information. Subsequently, it should be ensured that readers are able to identify which methodological choices are made and which kind of information is analyzed and to define the process of analysis. In the qualitative areas, another important point strongly related to the consistency of writing is related to the links between theoretical and practical dimensions. This point is particularly important in interpretation, which is the fourth step. In this respect, reliability, validity and usefulness are of great importance. The fifth step, which is the completion of Conceptual and theoretical studies, includes such steps as comparing the studies with other relevant researches and completing the missing issues. The final step is the preparation of the research report and making a digital or printed copy of all information collected.

According to Matotek et al. [48], Advantages of Qualitative Research are listed as: it reflects the "*reality*" of the special case, theory is easy to develop with the results, ensures that different factors are understood and applicability of the research results is higher.

As for its limitations; it is difficult for the subjects to express their experiences as is, and the analysis of the data also includes the prejudice of the individuals.

3.3. Quantitative Research

Besides qualitative research method the other main research method is quantitative method.

Quantitative methods are methods that focus on extracting data from existing data using proven statistical or general methods, the data are analyzed and numerical data are available. According to Matotek et al. [48], in the research developed by means of Quantitative methods, the progress is achieved by finding answers to the questions of how much? in what amount? how often? how common? and there are a number of methods.

1. Experimental Model

2. Descriptive Model
3. Relational Model
4. Causal-Comparative Model
5. Historical Method

Experimental Model: the differentiation dimension of the effects of the applications on two or more groups in terms of certain variables is examined. By means of statistical techniques, comparisons are made between groups and a result is obtained. Relational Model: used to examine whether there is a relationship between two or more variables. Causal-Comparative Model: it is aimed to compare the groups differentiating in terms of a certain variable. Historical Model: The situation of a past event and the effects of this event on the current situation are examined.

Advantages of Quantitative Research: generalizable results are produced, differentiation can be made between different groups, the accuracy of the theories is determined, relations within a particular structure is examined.
Its limitations are; it is difficult to get a perfect sample, it is difficult to collect enough data, perfect measurement conditions cannot always be achieved, the measurement tool also reflects the prejudice, it is not concerned with the data other than the model.

3.4. Mixed Research

According to Kajfez et al. [49], mixed method research has emerged as an accepted and rigorous method of inquiry. Mixed method research is defined as combining the qualitative and quantitative methods, approaches and concepts in a study or successive studies. Quantitative research model, which is more preferred in the past, has started to be replaced by qualitative research and then by mixed research, which is a mixed form of both qualitative and quantitative research. Indeed, the researcher strengthens the research with another method and obtains more realistic data by making the research more consistent and develops theory by increasing the generality of the research. Researchers, who want to use this method, should firstly dominate the qualitative and quantitative paradigm then skillfully blend these two methods.
Advantages of Mixed Research Method:

- It allows overcoming the weakness of working with a single method.
- It develops a detailed and advanced analysis by reaching richer details.

- Many benefits can be achieved by combining or linking quantitative and qualitative data collection methods.
- The advantage of mixed research is that it provides a better understanding of the literature.

Disadvantages of Mixed Research Method:

- Completing both qualitative and quantitative research can be difficult for a single researcher. This may require a research team, especially if two or more approaches are desired at the same time.
- It is more expensive.
- It requires more time.

3.5.Action Research

In Action research, the researhes try to overcome the real-world problems while concurently work on the problem solving experience as action researhes depend on experiments and they are emprical research methodologies. In the course of past twenty years, action research approach has been greatly used in social science and adopted in software engineering researches for informations systems. Baskerville [34] argues that, these four common characteristics are shared in action research studies: "*(1) An action and change orientation (2) A problem focus (3) An organic process involving systematic and sometimes iterative stages (4) Collaboration among participants*". The figure below shows the five basic steps of the action research:

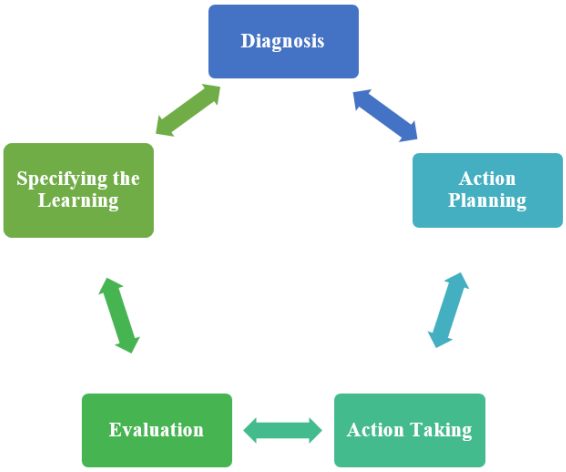


Figure 2: Action Research Cycle

Diagnosis: The primary and the most significant problems are identified in this step, which drive the organisation to make a change. The formulation of the problem domain of the organisation and the theorical assumptions about it are made in this step.

Action planning: The action plans for the identified problems in diagnosis step are mede in this step. We determine where we want to see the organisation at the end of the action researh and we create a list of actions in order to achive this.

Action taking: The planned actions in the previous step are executed in the action taking step.

Evaluating: The results of actiones taken in the action taking step are evaluated in this step and the theorical effects of the actiones are evaluated by the researhes to determine if they are accomplished.

Specify the learning: Whether or not the implementation is successful, the lessons learnt from the cycle has to be identified, which will be used for the determination of the prospective future cycles.

The reasons of choosing the action research in this study are to ensure that the problems and obstructions are determined, solved and eliminated in the studies that aims at developing and improving the existing product lifecycle compatibly for purpose of digitalizing and automatizing the processes and systems of AKGUN YAZILIM, and to ensure that the system, which will enable that all relevant stakeholders participate in these steps, cooperate, review the current situation, plan and take any actions and achieve the target, is included in the corporate memory and the existing system of the Company.

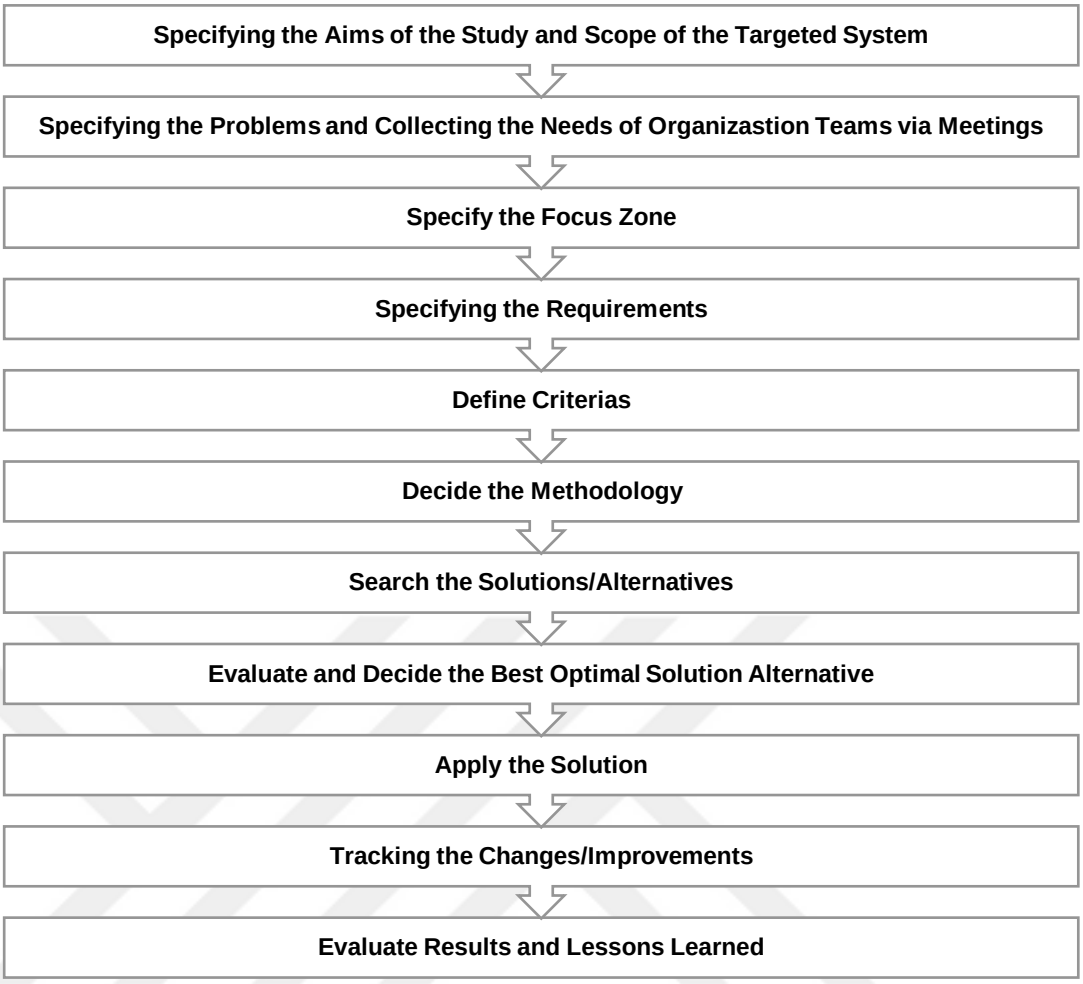


Figure 3: Research Map

Advantages of the action research:

- It includes any cooperative systematic reviews, which are designed and arranged previously to increase quality by means of research and interrogation;
- A study is conducted by one or more persons or groups to solve a problem;
- It includes any studies developed by people who implement them exactly;
- It enables use and implementation of the experiences;
- It ensures that the stakeholders give any advice each other and share any mutual comments;
- It requires being critical and open-minded; and
- It enables that the teams/implementers are professional in their profession.

The disadvantages of the action research:

- There is a risk that the process of monitoring the actions is kept longer than necessary;
- There is a risk that any waiting processes in expansion of the research results to the large masses may bring the study to a standstill in a long term; and
- Maybe any particular variables have not been checked or calculated in expansion of the research results.

CHAPTER 4

4. ACTION RESEARCH

4.1. Context and Motivation

The action research on the impact of ALM adoption has been carried out in a large-scale IT company, AKGÜN.

AKGÜN that considers information and technology as an essential for life was established in 1986 and provides solutions and a wide range of products from IT, healthcare, education, telecommunication, government, defense, and the private sector.

AKGÜN that has been in service on information systems and technologies for more than a quarter-century, has a wide array of project experience in areas such as in R&D, Manufacturing Software, System Integration, Training and Support Services, Network, Communication and Information Security, Technology Consulting, Medical IT Consulting Services and turn-key projects to its services and quality of products.

Since its establishment, the integrator company AKGÜN, that improves and renovates itself with its experiences and know-how, enhances the service quality that it holds and as a sustainable development company with growth capability, it has over 500 personnel together with over 80 researchers. Although its central office is in Ankara, it also has a branch at the Black Sea Technical University Technopark.

AKGÜN Plaza, central office located in Ankara, has a distinctive design, modern and separate construction and consists of a total of 6000 square meters, and acts as Turkish Ministry of Science, Industry and Technology Certified Research and Development Center. In addition, with its Azerbaijan and Kazakhistan branches, it continues research, development and maintenance work in line with its international vision and strategic plans.

With its solutions and 100% domestic products in more than 450 State Institutions and Organizations all over the country and abroad,

- Administrative Processes,
- Medical Processes,

- Financial Processes,
- Imaging / Image Processing,
- Decision Support Systems,
- Business Process Management,
- Evidence-Based Medicine Practice,
- Device Integration,
- Mobile Technologies,
- System Integration,
- Electronic Records and Document Management,
- Personnel Control and Monitoring Systems,
- Human Resource Management Systems,

The above mentioned services are provided synchronously.

AKGÜN has a perspective of providing quality products and services, it is a chief point of corporate management to be maintained in departments of Turkey and local structuring in which it is in service, it has NATIONAL SECRET / NATO SECRET Facility Security Clearances, CMMI Level 3, ISO / IEC 27001 Information Security Management System, ISO 9001, OHSAS 18001, ISO 13485, TS 13298, CE Marking certification and in addition, also has other certifications and accreditations of different sectors of activity.

AKGÜN supports fully compatible and end-to-end Digital Hospital concept thanks to its rich product and solution portfolio. It creates, implements and supports customized solutions by analyzing the needs of institutions. AKGÜN Hospital Information Management System applications are compatible with HIMSS Stage 6 and 7. In these terms, our teams perform adaptations according to the needs of institutions. Our company provides special solutions by using the advantages of Industry 4.0 within the digital age we live in.

Table 3: General Information About Action Research Organization

Number of Products	45
Number of Projects (capacity in a year)	20+
Personnel	500+
Engineers	100+
Project sizes	5-80 people
Programming Languages	C, C++, C# Java
OS	Windows, Linux, OS X
Development Process	Waterfall, Agile, Iterative

4.2. Action Research Cycle – ALM

Action research steps consist of diagnosis, action planning, action taking, evaluating and specifying the lessons learned.

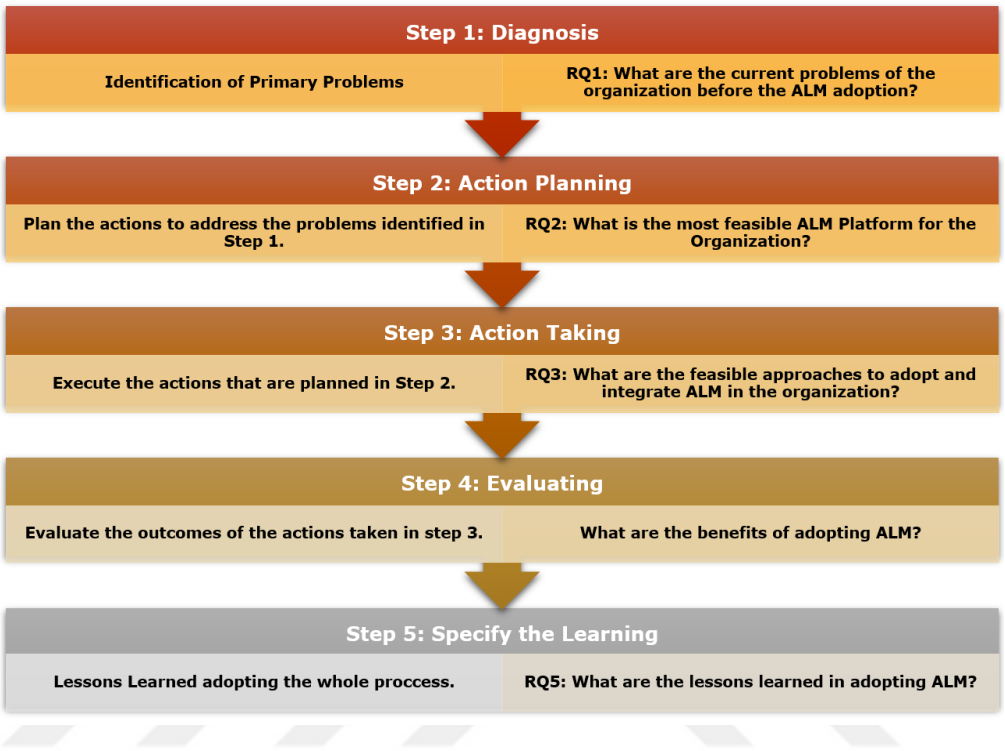


Figure 4: Action Research Steps and Research Questions

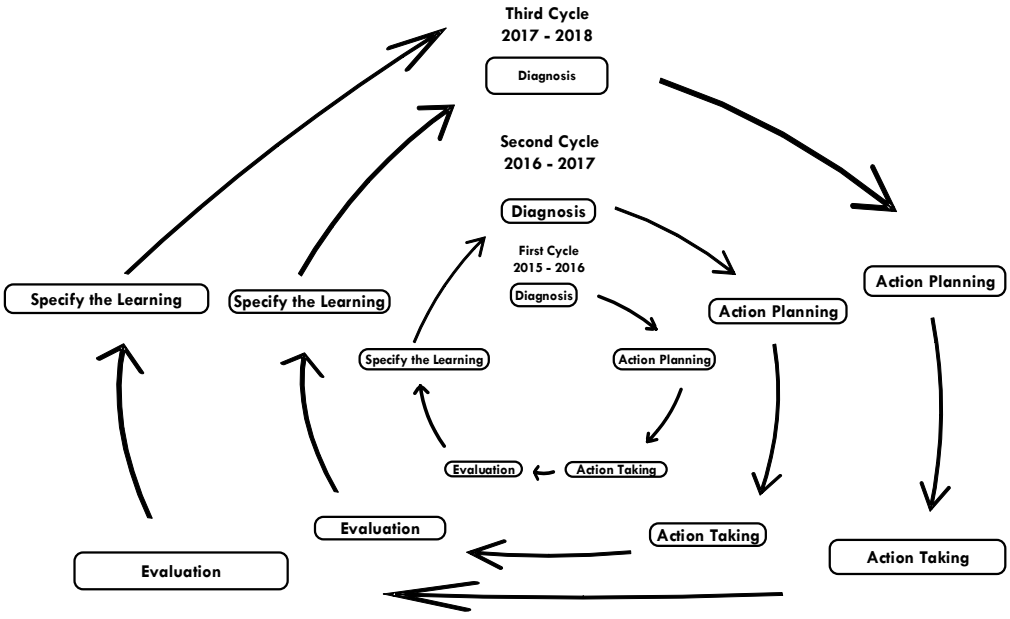


Figure 5: Sequential Action Research Cycles

4.3. First Action Research Cycle- ALM 1.0

4.3.1. Diagnosis

The following perspectives and questions have been considered in the process of these diagnosis:

Organisational perspective: How much the ALM adoption is supported or impeded by the organisation.

Process perspective: What software development processed are currently being used within the organisation and what are their existing problems or needs?

Tool perspective: What tools are currently in use? For the related lifecycle activities such as requirements and testing etc. what sort of adoption issues are we currently facing? What integration issues do we have in terms of tool integration?

To be able to carry out an effective diagnosis, the relevant groups and key personnel to carry out the works are identified within the company. The situation is assessed in terms of organizational, process and tool perspectives as a result of the reviews of the projects and interviews, meetings with 6 groups in total including Image Processing Group, Clinical Information Group, Web Technologies Group, Mobile Technologies Group, Integration Group, Decision Support Systems Group and the key personnel in these groups. The results of the examinations are as follows from each perspective;

4.3.1.1. Organizational Perspective

Project-based and strong matrix organization chart is used in the organization. Relevant persons from each group are assigned to the projects initiated according to the project's requirements and areas of expertise. Each project is assigned with a Project Manager by the Project Management Office. The assigned Project Manager is the most influential Executive until the closing of the project, and the entire project team is primarily responsible to the Project Manager in the first place. Until the closing of the project, reporting is made to the Project Manager rather than the Unit Manager. Upon closing the project, the team is disbanded and the relevant personnel may continue their responsibilities and work towards the Unit Manager. However, at times, namely in the joint use of resources considering the project and operation plan, there might be communication gaps and it may get difficult for the personnel to comply

with the reporting systematics for planning other than project and distinguish the Manager to report.

On the other hand, Analysis, Test and Software Quality Assurance Unit is a separate unit in the organization and the analysis, test and version distribution activities of all projects are among the duties of this department. In this respect, the inter-departmental integration and a fully integrated manner of work become important. In addition, as such issues as code review, error-free version, quality design are important in terms of ensuring the quality assurance of the software, and these issues need to be controlled and quality-assured before the release of the version, integration between the teams and processes is a very important point.

It is seen that some of the employees in the organization, use different tools on the basis of each project and section on the other hand, some groups still use their old working methods. With this study, it will be ensured that they are turned into a part of the system and that they are working harmoniously within the systematics.

4.3.1.2. Process Persective

The company has been developing small, medium and large-scale projects for the last 20 years and mainly implements waterfall, agile and iterative software lifecycles in research and development projects. In addition, AKGÜN Software has been awarded with CMMI Level 3 since 2013 and works in accordance with the CMMI model in the institutional, managerial, engineering and supports the implementation processes of CMMI model software lifecycle disciplines. PMI methodologies are applied in Project Management section.

The process implementation model of the company is schematized as follows;

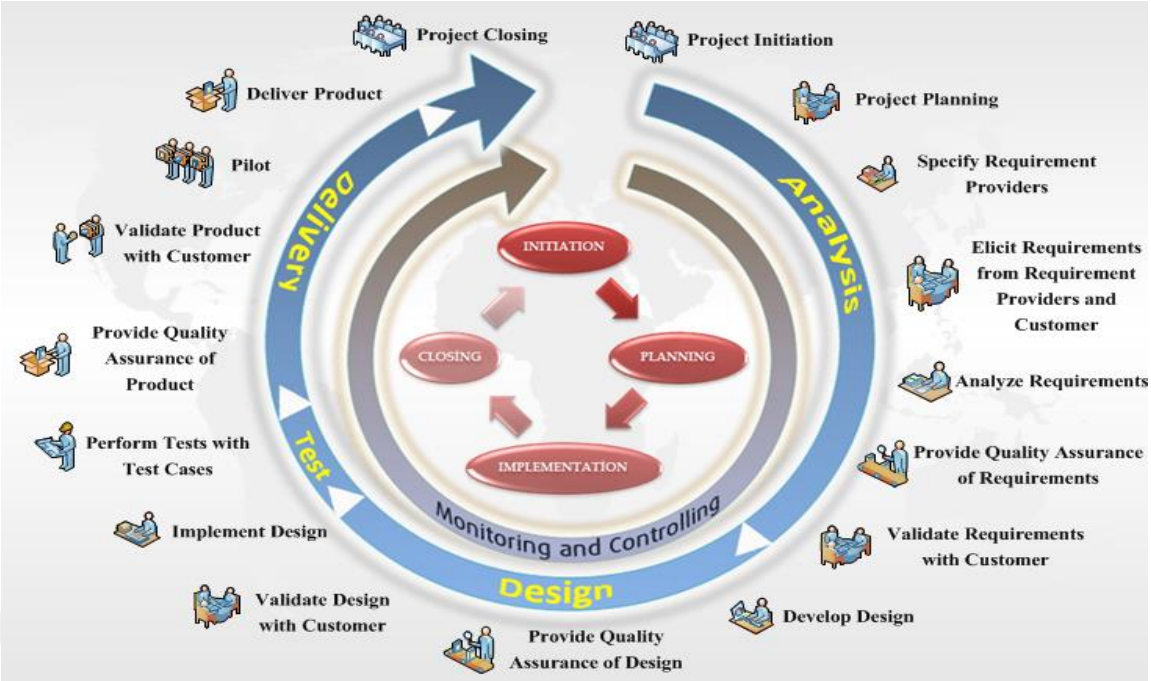


Figure 6: Project Lifecycle

The lifecycle that is schematized above is applied with different iterations according to the chosen software lifecycle. The application of each stage differs according to the software lifecycle. On the other hand, the AKGÜN SOFTWARE processes include process tailoring according to agile, iterative and waterfall software lifecycles and disciplines have been defined in order to establish the requirements of these processes and the relevant documentation in all projects and a continuous monitoring and control mechanism has been established for the implementation. In spite of this, in some projects incomplete or different process applications occur and the Quality Assurance Officers of the project ensure systematic adaptations by making instant and necessary interventions.

The documentation required to be prepared in each project has been determined and more limited and frequent documentation and process output are generated in agile software lifecycles. The Waterfall model has more strict documentation and rules compared to the agile processes.

4.3.1.3. Tool Perspective

The tools that support the processes implemented in the organization are found in more than one and in different ratios in each project. While some projects implement manual

documentation, other projects use different tools and different features of these tools for the same processes.

Table 4: Product Life-Cycle Tools

Lifecycle Activity	# of different tools in use
Requirements Management	5
Design Management	3
Integrated Development Environments	3
Test Management	4
Configuration Management	6
Change Management	5
Build Management	1
Knowledge Management	10
Project Management	2
Communication Management	4

4.3.2. Action Planning

Within the scope of the planning of the actions, firstly the focus is on the solution of the problems that are determined in the diagnosis phase and are identified as a result of the meetings held.

One of the biggest challenges in adapting ALM is; from the organizational perspective, the resistance of the old teams and other teams working in the project to change the working culture. To this end, a series of meetings and trainings have been initiated and our resolution in this regard has been explained to all teams. Our goals and determination of our management in this transition process has been underlined in particular. Several meetings have been planned on this issue and the benefits of this transition have been repeatedly told, stressed and we have tried to convince the teams.

In the action plan, we can evaluate the process and tool perspective together. Correct tool selection and adaptation are both linked.

Several alternative tools have been researched for use in different areas within the company. One of the most important criteria in choosing a tool is to be compatible with our CMMI processes and to ensure traceability between projects and processes.

The requirements have been determined in the first place to select the tools and create the best tool set, the best possible alternatives have been identified, and then interviews with appropriate suppliers and POC studies have been carried out.

A committee has been established within the company to carry out the action planning.

First of all, expectations from the tool, namely user needs, have been identified. The user needs identified will also be an input to the tool suppliers' assessment and technical specifications of contracts with suppliers.

The committee established have completed the below stages in the action planning phase.

Analysis: At this stage, user requirements and tools have been analyzed in the first place.

Assessment: The assessment has been made according to the criteria determined by the assessment committee for the tools-materials researched. In the assessment stage, the scoring system has been applied and the selection of the alternative, which has the highest score, is planned.

Operation: At this stage, the assessment of the suppliers and the selection of the most appropriate and fulfilling alternative and progress in the necessary studies will be ensured.

4.3.3. Action Taking

At this stage, primarily the analysis of the tools was carried out. In this context, many ALM tools have been examined.

4.3.3.1. Analysis

At this stage, the requirements of the system have been developed. During the development of the system requirements, the company's defined corporate processes have been examined in the first place. In the first stage of the action research, it is planned to deal with the analysis, design, coding and test sections that form the software lifecycle and technical processes.

To ensure more effective management of action research studies within the scope of ALM and achievement of its goal, it has been decided to select a "*requirement and test management*" tool, which is required to be used to operate the analysis and test

processes already in use in a more efficient and effective manner as well as the needs such as change management, requirements management, dual traceability etc., which are included in the CMMI requirements, in the first cycle. Action Research Cycle ALM 1.0 study was carried out in 2016, primarily the requirements, design and test management and traceability of these have been the focus.

"*Requirements Development Process*", "*Requirements Management Process*", "*Configuration Management Process*" and "*Testing Process*" of AKGÜN SOFTWARE have been referred in the identification of system and user needs. Based on this, the user needs have been formed as follows and the relevant project teams have been finalized as a result of the interviews and reconciliations.

Table 5: List of the System Requirements Identified Within the Scope of ALM 1.0

No	Requirement	Relevant Process
1.	Various rules should be defined on the areas defined for requirements or tests. For example; a. If the value of the X field is 'a', no data can be entered in the field Y. b. If the value of the X field is 'a', the user Ahmet cannot update/view this record.	Requirements Management, Test
2.	Review (peer review) processes should be managed through tool. In this context, the tool should keep information as to the fact that the requirements and tests are in a 'reviewable' condition, the appointment of the personnel to review should be managed and the comments of these reviewers should be stored on the tool.	Requirements Management, Test
3.	Test steps and the requirements should be associated.	Test
4.	Each step of the use state flow (scenario) should be associated with other requirements separately.	Requirements Management
5.	Different types should be defined for the relationship established between the requirements-requirements (trace). The relationship type and description information should be stored for each relationship. Definition and or removal of relations should be recorded in the requirement history.	Requirements Management
6.	Records of meetings with the requirements provider, approval mechanisms and planned actions should be recorded.	Requirements Development
7.	Prioritization of the requirements should be done automatically by running the formula defined on the tool after the user has scored certain criteria.	Requirements Management
8.	Templates should be defined on the tool so that requirement and test documents can be exported. These should be available in MSWord, MSExcel and PDF formats.	Requirements Management, Test

No	Requirement	Relevant Process
9.	The tool should automatically import the requirements and test scenarios from the analysis and test document templates already being used within the Company.	Requirements Management, Test
10.	Image files should also be stored within the requirements. Flow diagrams developed within the scope of analysis activities should also be drawn on the tool.	Requirements Development
11.	Baseline times should be recorded in the system.	Requirements Management, Test
12.	Traceability matrix report compliant with the specified filters (requirement, test) should be received. Printout of this report should be available in MSWord, MSExcel and PDF format, when and where required.	Requirements Management, Test
13.	All fields such as requirements status, version number, priority level etc. should be reported in tabular form.	Requirements Management
14.	Baseline should be obtained on the requirements and test scenarios and no changes should be allowed after baseline.	Requirements Management, Test
15.	When there is a change in requirements, the tool should alert the relevant users to check other related requirements and tests.	Requirements Management, Test
16.	Logs of changes to the system (user, date, time, change reason etc.) should be kept and be accessible.	Requirements Management, Test
17.	Test results should be stored on the tool. In the test results, the status entry for each step should be possible. Again, for each step, the success or failure status of the test result should be entered and reported. Errors detected as a result of the tests should recorded on the tool along with the detailed information and error records should be dually synchronized with Atlassian Jira tool, which is already being used.	Test
18.	Before carrying out the tests, test scenarios prepared and steps should be reviewed.	Test
19.	Test type and test level information should be stored for the test scenarios prepared. For example; functional, performance, integration, system etc.	Test
20.	System should digitally and automatically calculate and report the total number of scenarios, the total number of steps, the total number of successfully-completed steps, the number of critical errors, the number of very serious errors, the number of serious errors, the number of intermediate errors and the number of slight errors.	Test

The application development lifecycle management (ADLM) focuses on the tool market, software development lifecycle (SDLC) planning and governance activities. ADLM products focus on the development part of the life of an

application. In recent times, research criteria have been changed to measure the Agile development and DevOps support in a robust manner.

The key elements of an ADLM solution include:

- Software Requirements Definition and Management
- Software Change and Configuration Management
- Item Management by Focusing on Software Planning, Agile Planning
- Quality Management including Error Management
- Reporting Work Flow Integration to Version Management

The tools and equipment within the scope of ALM, which can meet these needs, have been researched and it is seen that there are many alternatives developed in this regard.

In the case of software projects, specific application lifecycle management tools are used to ensure successful project management and successful results. Recent studies and research have increased the focus on Agile and DevOps applications in application development lifecycle management.

A review of the Gartner Survey has been conducted to assist ALM providers, application development managers, and other IT leaders to select existing technology partners in the industry where they work [32]. This research is based on the most preferred application lifecycle management tools used today and provides information about the tools. A world-wide enterprise that researches technology tools, Gartner uses a method called Magic Quadrant for these operations. The answer to which players in the market are superior to their competitors and in which areas they are superior is looked for by means of this method. By means of the method, all tools in any market can be compared. Research conducted by Gartner using this method is published on their own website [32].

Figure 7: Magic Quadrant Template

Detailed information on the graphic provided with the methodology is given below; [32]

Leaders, represent the products that work very efficiently and have a good future vision.

Visionaries represent the respectively low-efficiency alternatives that know where the market is going, change the rules of the market accordingly and take action.

Niche Players, represent the alternatives that have succeeded by focusing on a small segment and do not target any innovation.

Challengers, represent the alternatives that are well capable but do not recognize where the market is going properly.

The evaluation criteria are separated as vertical axis and horizontal axis and are detailed below;

- **Working Performance (Vertical Axis)**

- Product and Service (High Importance)
- General Vitality (Moderate Importance)
- Sales Price (Low Importance)
- Market Eagerness (Moderate Importance)
- Marketing Performance (Moderate Importance)
- Customer Experience (High Importance)
- Operations (Low Importance)

- **Vision (Horizontal Axis)**
  - Market Perception (High Importance)
  - Marketing Strategy (Low Importance)
  - Sales Strategy (Low Importance)
  - Offered Product Strategy (Low Importance)
  - Business model
  - Vertical Positioning Strategy
  - Innovation Capability
  - Graphic Strategy

In obtaining these results, various studies and research have been taken as basis. An important element is the use of online surveys of customers of the involved vendors. In addition, Gartner has analyzed the customer call data and information from various social media sites has been used to understand the effectiveness of the application.

The research report published on the Gartner official website has a detailed list of the tools used [33]. Companies in this list: Atlassian , IBM, Microsoft, Rally (now CA Technologies), Micro Focus, VersionOne (CollabNet), Jama Software, Perforce Software, CA Technologies, Rocket Software, CollabNet, PTC, Siemens, SUSE, Kovair Software, Beesion, Inflectra, Digite.

The products of the companies in the list are listed in the table and in this context; the information of ALM provider company names, products, tool score, license type and tool's appraisal score is detailed.

Table 6: ALM 1.0 Companies Providing the Platform, Product Names, License Type and Access Information

Company Name	Tool Name	Licence Type	Point	Reference
Atlassian	Bamboo	Commercial	4.3	https://www.atlassian.com/software/bamboo
Atlassian	Bitbucket	Commercial	4.3	https://www.atlassian.com/software/bitbucket
Atlassian	Clover	Commercial	4.3	https://www.atlassian.com/software/clover
Atlassian	Crucible	Commercial	4.3	https://www.atlassian.com/software/crucible
Atlassian	FishEye	Commercial	4.3	https://www.atlassian.com/software/fisheye
Atlassian	JIRA Software	Commercial	4.3	https://www.atlassian.com/software/jira
IBM	IBM Cloud - PaaS (Bluemix)	Commercial	3,8	https://www.ibm.com/cloud-computing/bluemix/node/2409
IBM	IBM Rational ALM	Commercial	3,8	https://www-01.ibm.com/software/rational/alm/
Jama Software	Jama	Commercial	4.2	https://www.jamasoftware.com/solutions/requirements-management/
CA Technologies	AgileCentral	Commercial	4.3	https://www.ca.com/us/products/ca-agile-central.html
Siemens	Polarion ALM	Open Source	3.3	https://www.plm.automation.siemens.com/store/en-us/trial/polarion-alm-download.html#ACTION=3856996352
Perforce Software	Perforce Software Helix	Open Source	4.3	https://www.perforce.com/
PTC	PTC Integrity	Commercial	4.3	https://www.ptc.com/en/products/plm/plm-products/integrity-lifecycle-manager
Kovair Software	Kovair ALM Studio	Open Source	5.0	https://www.kovair.com/
SUSE	SUSE Studio	Commercial	4.5	https://www.suse.com/products/susestudio/
Inflectra	SpiraTeam	Commercial	4.0	https://www.inflectra.com/SpiraTeam/
Digite	SwiftALM	Open Source	3.0	https://www.digite.com/swiftenterprise/

Company Name	Tool Name	Licence Type	Point	Reference
Microsoft	Microsoft Developer Network	Commercial	4.2	https://msdn.microsoft.com/tr-tr/dn308572.aspx
Microsoft	Team Foundation Server	Commercial	4.2	https://visualstudio.microsoft.com/tr/tfs/
Rocket Software	Rocket Aldon	Commercial	4.4	https://www.rocketsoftware.com/products/rocket-aldon
Rocket Software	Rocket Software	Commercial	4.4	https://www.rocketsoftware.com/product-categories/application-lifecycle-management-and-devops
CollabNet	CollabNet CloudForge	Commercial	4.3	http://www.cloudforge.com/
CollabNet	CollabNet TeamForge	Commercial	4.3	https://www.collab.net/products/teamforge-alm
Micro Focus	AccuRev	Commercial	4.0	https://www.microfocus.com/products/change-management/accurev/
Micro Focus	Caliber	Commercial	4.0	https://www.microfocus.com/products/requirements-management/caliber/
Micro Focus	Micro Focus (HPE Software) Application Lifecycle Management	Commercial	4.0	https://www.microfocus.com/about/press-room/article/2017/micro-focus-completes-merger-with-hpe-software/
Micro Focus	Micro Focus (HPE Software) Application Lifecycle Management Octane	Commercial	4.0	https://www.microfocus.com/about/press-room/article/2017/micro-focus-completes-merger-with-hpe-software/
Micro Focus	Silk Central	Commercial	4.0	https://www.microfocus.com/products/silk-portfolio/silk-central/

Company Name	Tool Name	Licence Type	Point	Reference
Micro Focus	StarTeam	Commercial	4.0	https://www.microfocus.com/products/change-management/starteam/

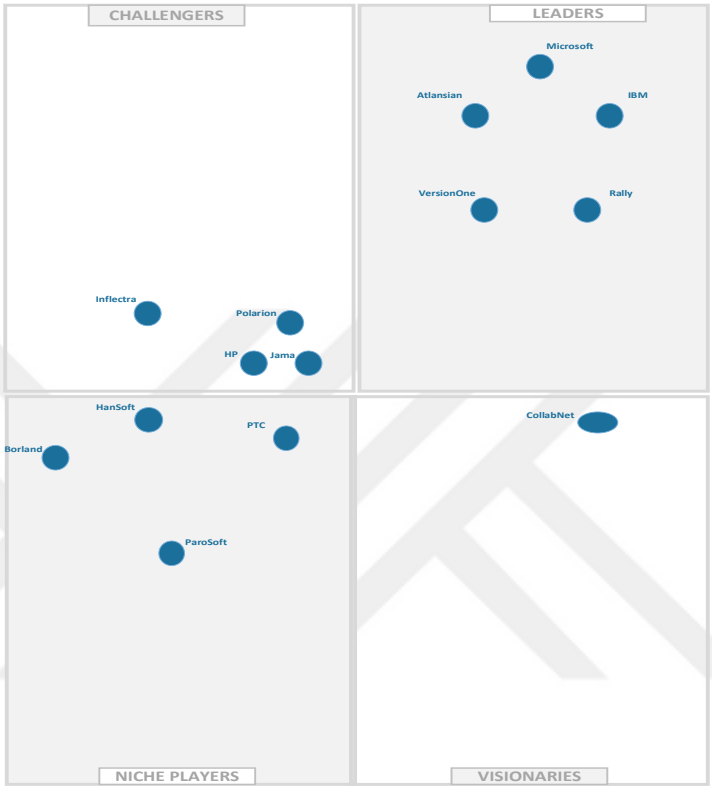


Figure 8: Magic Quadrant for Application Development Life Cycle Management [32]

According to the results of the research conducted by Gartner and as can be seen in the above graphic, Microsoft, IBM and Atlassian are listed among the most popular ALM Tools [32].

4.3.3.2. Assessment and Acquisition

Based on the assumption that taking all of the abovementioned tools, which have been obtained during the analysis stage, among the alternatives may cause unnecessary studies and loss of labor force and at the same time, not all of them will meet the requirements and quality, these tools have firstly been researched on the internet and, if necessary, demonstrative versions have been downloaded and examined. The tools that are not qualified to meet our requirements have been identified and eliminated.

As a result of the researches and examinations, it has been decided to consider the following tools, which are likely to meet the user requirements.

- IBM Jazz Platform
- HP Application Lifecycle Management
- MS Team Foundation Server
- Enterprise Tester
- JAMA

Each alternative remaining in the second list has been examined in detail in itself and the strengths, weaknesses, opportunities and risks in the table below are identified as a result of the preliminary research and evaluation on the tools.

Table 7: Comparison Table for the Strengths, Weaknesses, Risks and Opportunities of the ALM Tools Evaluated

Tool Name	Strengths	Weaknesses	Opportunities	Risks
IBM JAZZ	-Flexible and customizable tool -IBM brand and strong partner firm	-Management difficulties and effort requirements -Consulting needs during initial installation -High cost -A platform consisting of different tools	-Production phase can be incorporated into automation integrally within the same platform in the future.	- Decrease in the use ratio and efficiency of the tool unless necessary effort for maintenance, care is allocated. - The tool becomes idle unless additional costs such as annual maintenance fees and license upgrade are met.
HP	-Flexible and customizable tool -HP brand and strong partner firm -Licensing model -Existing experience in management and configuration	-High cost	-Integrated operation with very powerful other tools of HP in the field of test	- The tool becomes idle unless additional costs such as annual maintenance fees and license upgrade are met.
TFS	-Free right to use	-Development and test driven	-Not observed.	-Possibility that the burden of the tool will

Tool Name	Strengths	Weaknesses	Opportunities	Risks
		different methodology -Requirements management is very weak. -Suitable for the projects using .NET. -Restricted features with different technologies.		outgrow its advantages.
JAMA	-Ease of use and management	-High cost	-Not observed.	-Reliability and continuity of the firm -Failure to provide instant and on-site support - The tool becomes idle unless additional costs such as annual maintenance fees and license upgrade are met.
ET	-Low cost	-Simple and limited in terms of capabilities -More focused on test management. -Particularly weak in terms of requirements management.	-Opportunity for integration with automatic test tools such as Selenium.	Possibility of finding the tool insufficient after a while.

As a result of this research, it has been decided to detail the works within the scope of each tool. The following action plan has been developed during the detailed study of each tool;

- Determination of the user license requirements and usage frequencies,
- Obtaining contact information of ALM tool suppliers,
- Installation of demonstration versions of the tools,
- Examining of the tools by the technical evaluation committee of our company in the first place,
- Submitting a list of our user needs to the ALM suppliers and requesting their explanations for meeting each one of our needs,
- Receiving the license-based price offers from the ALM suppliers,
- Inviting the ALM supplier companies to the demonstration meetings and ensuring that demonstrations on the tool are made by the firms by holding meetings in this scope and evaluating the needs and solution methods in the form of questions-answers

Scoring Alternatives and Evaluation Systematic:

In our organization, decision analysis is made at the stage of evaluation of alternatives for each decision.

Decision-making / selection criteria are determined during the evaluation of alternatives. The appropriate selection criteria and evaluation methods are used to select the appropriate alternatives. At least 3 alternatives are determined. Criteria; plans and targets compliance, quality expectancy, customer satisfaction, delivery time, warranty period, training support, technical criteria, user requirements, technological compatibility etc. criteria and considerations are formed.

The selection criteria are rated as 1 lowest and 5 highest by order of importance. The stage of rating the selection criteria is determined by the common opinion of the people, who form the alternatives, and the decision makers. The current situation and conditions, in which the subject to be decided is, are also taken into consideration. At the stage of determining the selection criteria, whether the criteria are included in the evaluation or not and the reasons are also evaluated. In this case, a check mark is added next to the criteria included in the evaluation, and why it is selected is indicated. A description of why the criteria are not included in the evaluation is also indicated.

The determined alternatives are taken into evaluation according to each selection criteria that is formed. The person, who formed the alternatives, may refer to the scoring systematic, old experiences, comparison with similar ones serving the same purpose (survey and research results, promotional presentations), cost analysis techniques, prototypes, risk analysis, business opportunities assessments and tests among the alternative evaluation methods.

Selection Criteria and Scoring Method is commonly used as the evaluation method in evaluation and the alternative with the highest score is selected. If there are other evaluation methods in besides the scoring systematic, details are explained.

When determining the selection criteria and evaluating alternatives, the level importance of the criteria should be taken into consideration. 5 is the highest level of importance, whereas 1 is the lowest level of importance. The scoring is made between 1-5.

Then, alternative's score of meeting the criteria is scored again according to the level of importance given for each criteria. The level of importance of the criteria and the alternative's score of meeting the criteria are multiplied and the score for that criteria for the alternative is obtained. This method is applied to all determined criteria and the scores are summed. The alternative with the highest score is presented as a suggestion. Within the scope of the approved alternative, the required supplier and contract management and other administrative processes are started and technical works begin.

Table 8: Rating the Alternative's Score for Meeting Criteria and Descriptions

Score	Level of Meeting Criteria	Description
1	Inadequate	The alternative does not meet this criteria.
2	Low	The alternative meets this criteria on a low level.
3	Average	The alternative meets this criteria on an average level.
4	High	The alternative meets this criteria on a high level.
5	Very High	The alternative meets this criteria completely.

During the evaluation and selection of the alternative ALM tools; selection criteria have been determined for these tools and then criteria have been grouped in two different ways as organizational and technical;

Table 9: Alternative ALM Tools Selection Criteria List

No	Criteria Description	Criteria Group
1.	Budget and time of the organization	Organizational Criteria
2.	Mission, plan and goals of the organization	
3.	Time concept of the organization	
4.	Quality expectation and customer satisfaction of the organization	
5.	Amount of effort exerted	
6.	Possible risks	
7.	Technology parallelism	
8.	Expert opinions, references and performance	
9.	Is Training Support available? (learnability, continuity of use, durability etc.)	
10.	Payment Term	
11.	Defining different requirement types	Technical Criteria
12.	Defining custom fields	
13.	Associating different requirement types with custom fields	
14.	Establishing trace between the usage status steps and other requirement tests	
15.	Establishing trace between the test steps and the requirements	
16.	Warning for the associated requirement or tests when a requirement is updated	
17.	Recording of the history of the requirement and test updates	
18.	Creating baseline and changing baseline content	
19.	Display of relevant requirement and tests for impact analysis	
20.	Reporting skills	
21.	Flexibility and customization opportunities	
22.	Ratio of compliance to our processes	
23.	Potential of including design phase and relevant process(es)	
24.	Potential of including coding phase and relevant process(es).	

The weight points of each selection criterion and the score for tools' fulfillment of our selection criteria are given.

The score of each relevant tool according to the each criteria has been calculated using the Tool Weight Rating * Score for the Alternative Tool's Fulfillment of the Selection Criteria formula.

According to the selection criteria identified as a result of the evaluations made using the scoring technique, it has been decided to choose the IBM JAZZ ALM tool, which has the highest total score.

After having submitted the evaluation results and approval of the MANAGEMENT, procurement processes have been initiated. After this stage, the following actions have been completed by meeting with the supplier;

- Drafting of the Technical Specification,
- Preparation of the contract,
- Establishment of the installation and training plans for the commissioning of the tool,
- Determining and contracting the warranty process and licensing upgrade requirements,
- Mutual signing of the contract.

4.3.3.3. Base Installation, Customization and Development and Adoption of Pilot Projects

Upon signing the contract with the IBM JAZZ Supplier, installation and adaptation works started within 3 days.

Within the scope of JAZZ IBM, there are 4 different sub-component tools that meet our needs and installation and inter-project integration of each one of these have been carried out in the installation and customization process. The subcomponents purchased under the IBM JAZZ integrated ALM tool are as follows;

- RRC - Rational Requirement Management
- RQM - Rational Quality Management
- RTC - Rational Team Concert

- RCCM - Rational Change and Configuration Management

As the first stage of the installation processes, the users have been transferred to the system by means of the installations of the latest version of the tool and LDAP integrations. The activities carried out along with and concurrently with the installations are listed below;

Within the scope of transferring the processes of AKGÜN SOFTWARE to JAZZ (RRC, RTC, RQM and RCCM);

1. Clarification of work schedule,
2. Ensuring access of all relevant users to the product by completing the LDAP connection,
3. Development of the requirements and test projects for the transfer of existing requirements processes to JAZZ upon being analyzed,
4. Determining the artifact types within the context of requirements management,
5. Continuing to create the artifact types over requirements management and developing the folder structure for the modules,
6. Creation of the templates used by AKGÜN SOFTWARE for requirement documents and reports within the tool,
7. Developing project and module templates on the RRC side and benefitting from the templates in the development of new project/module,
8. Initiating the transfer of this process to the RQM side by discussing the current test process on the RQM side,
9. Identifying the types of links required to ensure traceability for requirements, defining the types of links to create links for all requirements,
10. Review and update of the existing templates according to defined link types,
11. Performance of RTC installation and necessary configurations,
12. Continuing the process of transferring the corporate processes to the system and establishing the flows, by discussing the current test process on the RQM side,
13. In accordance with the institutional processes used at AKGÜN SOFTWARE, customization and definition of testing plans, test environments etc. accordingly,
14. Continuing the transfer of test plans and test scenarios on the RQM side, transferring the structure used in AKGÜN to the system,
15. Reviewing the project and module templates created by the RRC and making additions and then reviewing the final version of the templates and performing the final checks,

16. Reviewing the test scenario, test plan and test step templates created by the RQM and making additions and then finalizing the templates,
17. Completing the development of Analysis-Design, Test and Error work flow and document templates,
18. Establishing the Lifecycle project (integrated project with analysis, design, test, error management), then creating a part of the module projects on the RRC side and establishing the connection with the lifecycle project,
19. Continuing to develop the module projects on RRC side and establish the connections with the lifecycle project, last arrangements with the link types,
20. Assigning the persons to projects created within the tool, defining the roles,
21. Providing controls for end users before commissioning the system,
22. Transfer started through the Judicial Report Book Analysis Document, which has been selected as a pilot,
23. Continuing to the transfer of the Judicial Report Book Analysis, Design and Test Documents selected as pilot,
24. Establishing the Corporate Mobile Health Information System Project and adding the related persons to the project and defining their roles,
25. To initiate the transfer of the existing projects to IBM JAZZ, holding a practical meeting with the relevant team representatives to show how the transfers will be made,
26. Analysis-Design, Test and Admin User Documents have been reviewed with the relevant personnel and groups and the missing/erroneous parts have been corrected.

The duration of the commissioning of the system within the scope of these activities including the installations is 4 months and we have worked with the supplier firm for 3 days of the week. As per our institutional processes, works of each working day have been recorded in the signed forms against Service Delivery Minutes.

Admin and user guides have been developed to be used within the scope of IBM JAZZ, the relevant teams have received training and their certificates at a ceremony upon completing the training successfully. In this way the use of the system is encouraged by means of motivation and advances have been made to achieve our goal of automation of the development benches.

The last revisions and adaptations have been carried out before the tool goes live within the scope of all projects;

- Listening to users' questions about products, analyzing them,

- Installation of the fix (ifix10) released by IBM to the system,
- At the beginning of the study, a meeting with the users has been convened for the questions of the users about RRC and RQM. During the meeting, information has been provided on the requests that can be met with the current features of the product.
- Apart from this, information on the features that are not available with the current product but requested by the users has been collected and the process of conveying these to IBM as a new request has been initiated. However, a clear calendar has not been sent by IBM for the fulfillment of our requests.

As a result, IBM JAZZ's Requirements Management, Quality Management, and Error Management add-ons have been tailored to our CMMI processes and project management methodologies within the targeted plan and scope in a qualified manner. Information on adaptations to the tool is summarized in the table below;

Table 10: Statistics of the IBM JAZZ Configuration

# of Project Templates	4
# of Document Templates	4
# of Artifact Types	32
# of Artifact Templates	26
# of Document Types	2
# of Tracebility Types	39

4.3.3.4. Going Live

Tool has been made ready for the use of the entire company and experiences in the pilot have been shared with the other projects.

Documents under the projects have been transferred into the ALM tool and the list of the transmitted documents is as follows. IBM JAZZ Requirements, Quality, Error and Configuration Management products, which are licensed applications of AKGÜN SOFTWARE, are used as configuration management system for the following documents of projects;

- Analysis Document

- Design Document
- Change Request and Impact Analysis Form
- Requirements Traceability Matrix
- Requirements Prioritization Template
- Test Plan Template
- Test Scenarios Template
- Test Result Report Template

Within the scope of the activities carried out to go live, a presentation has been made to the MANAGEMENT and Table 10 has also been addressed in the presentation. During the meeting held, it has been decided to plan the transfer of all the projects to IBM JAZZ.

Table 11: Usage Statistics of the ALM Platform

General Usage	
#Projects and Modules	111
#Users	104
Total # of Work Items/Artifacts	39.124
Revisions of Work Items/	2.254

The space required for the installation of IBM JAZZ is reserved on the virtual server within AKGÜN SOFTWARE. This space is set to cover the entire system not as project or unit-based. IBM JAZZ has been installed by the IBM Partner in this space and this area has been checked periodically. When this allocated capacity becomes insufficient in time, the space on the server is expanded after the physical backup is taken. Backup of IBM JAZZ is realized in two steps;

- Database backup is carried out automatically on a daily basis thanks to a task written on the tool of the database itself.
- File System backup is carried out manually. Changes made to the file system on the virtual server are compressed and backed up daily.

It is ensured after each backup that these backups are taken correctly and in a working manner by using automatic tools.

4.3.4. Evaluating

Table 11 shows the benefits of the organizational perspective with transition to ALM tool as before and after ALM. The additional training costs and the lost labor due to the training are included in this table. Various tools and templates have been acquired prior to ALM adoption on a regular basis and educational process within the company needed to be organized both as the processes have been updated and within the scope of the orientation trainings for the new personnel. The tool trainings and the time spent during these trainings, which could be spent on development, have resulted in a significat cost. With the adoption of ALM 1.0 this has changed as only one type of tool training had to be organised. This helped the company to save on the training cost and the labor loss as the training was only necessary at the beginning and there is no longer need for time allocation for the training of additional tools.

Table 12: Organization Benefits

Factor	**Before ALM 2.0 application**	**After ALM 2.0 Application**
Additional # of Training Sessions	125	78
Workforce Loss because of Training (in a year)	250 man day	45 man day

Table 12 shows The benefits of the production, external and internal data transfer costs, project commencement speed, process audit and decision support durations. The project start-up costs has been greatly reduced by the ALM platform's integrated work environment. Prior to the adoption, purchasing, installing, customising and integrating the tools that were specific to the projects used to take a considerable time and effort. With the ALM 1.0 platform, this can be realized much more quickly and easily because it already provides an integrated working environment. Some basic configurations for product initialization were needed as opposed to before, including the project name, members of the project and the system parameters initialization.

Table 13: Production Benefits

Factor	Before ALM 2.0 application	After ALM 2.0 Application
Project Initiation Speed	3 months	3 hours
Traceability Management	60 man day	0 man day
Process Audit Time	18 man day	4 man day

The adoption of ALM 1.0 was obviously highly beneficial for the company as can be seen in the above tables.

The transition of new projects to the ALM 1.0 is not very concerning, however, for our existing project a transition plan is a must. This can be challenging as each project may have different requirements. Therefore, an automatically implementable common migration plan have been created for all projects to be tranferred with no issues. Some project members were resisting to this migration as they did not want to change the tools they were currently using for their ongoing projects.

The software is used for lifecycle requirements, design, testing, error management and change management. IBM JAZZ provides basic services that allow group applications to work together as a single logical server. The IBM JAZZ product ensures that two important parties of Quality Management (QM) and Requirements Management (RM) are managed and followed. In addition, applications stored with the same IBM JAZZ products communicate with each other. Integration is achieved in this way. After associating project areas in different applications, artifacts in those project areas can be linked. For example, the requirements in an RM project area can be linked to work items in the project area of Change and Configuration Management (CCM), requirements and work items can be linked to test plans and test scenarios in a QM project area.

JAZZ IBM provides only one synchronized database shared by the server and all registered applications. Directly centralized JAZZ Team Server can create and manage user accounts and assign licenses to users in applications that are saved on the Web client or server. User records in the pool can also be synchronized with user records in a Lightweight Directory Access Protocol (LDAP) directory.

In each application, the teams work within the space allocated for the project area. A project area defines the items to be delivered, the team structure, operations, and timing of the project. Project managers are responsible for creating and managing project areas.

IBM JAZZ Team Server includes a reporting component that helps you track the actions, behavior, and progress of a team or project. Reports can provide an effective decision-making by allowing information to be seen at a glance. The report component contains a data warehouse to store read-only history and aggregated data. This data warehouse is in the best form for efficient queries and fast response times. A report engine generates reports by accessing the data stored in the data warehouse.

Dashboards are a visual web interface that can be edited to display high-level information about the status of the project. It provides easy access to complete information from a dashboard. Dashboards visually present how data is integrated into multiple IBM JAZZ components.

Application Lifecycle Management and systematic management of software lifecycle processes according to international standards and methodologies are very important for project/business monitoring. Process outputs guide the people, who are interested in the issue, when there is missing information or an open issue and present an effective and efficient working systematic. Considering the importance of this, it is of great importance that corporate governance is effectively implemented by the project managers and other relevant stakeholders of the software lifecycle within the scope of the institutional company.

4.3.5. Specifying the Learning

The integration and the compability of the approved tools and processes causes significant issues for the company, which was revealed by the examinations during the diagnosis phase. ALM studies could only be supported upto a certain extend with Matrix Structure Organisation. Communication, analysis and the overall efficiency has been affected by the diverse tools utilization and process uncertainties. The Diagnosis phase has been very useful in highlighting all major issues and convincing the managers and developers to resolve existing problems and reach a consensus to prevent future problems. In order to address the issues from the institutional, tool and process point of view, a systematic approach was adopted while dealing with the

activities during the action planning. This approach gave the company a confidence as it seemed that the problems will be resolved.

Throughout the action research, lessons learned and significant comprehensions documentation have been created. Our action research study has enabled us to integrate the ALM requirements into the corporate system by taking an integrated review and ultimately addressing them. Software development practices in the company have been better understood and comprehended and the team has acquired a significant amount of information about the best tool kits and ALM concepts for each process area.

Above all, it is seen that there are requirements and complex requirements for integration within the scope of determining the identified customer needs and the tools, which help the lifecycle activities are not at the level to meet all our needs. We found out that some of the problems we face can be very diffucult to deal with. As a result of our research, we have identified some tools that can simultaniously support integration and satisfy the necessary lifecycle activity needs.

The necessary solutions and the integration issues were summarised in the tender dossier. During the bidding process, the considerable amount of suppliers were withdrawn as they saw themselves not fit. With the remaining suppliers, the development of economically viable and sustainable solutions continued. From this process, we have learned that the problems identified are difficult and that a complete solution cannot be developed by focusing only on the best of certain tools. This was indeed an insight that was not widely discussed in the literature at that time.

In the end, it has been proved that the adoption of the ALM 1.0 platform was the right decision according to the series of actions taken. Furthermore, a sustainable solution for the identified problems in the diagnostic process have been implemented with the help of the experiences during the adoption of this platform.

We decided to implement a gradual approach to adopting ALM to reduce risks. For this purpose, we evaluated the implementation of pilot projects in various sections. This has not only helped to get feedback to customize the platform but also to directly include the engineers in the general transition process. It was observed that non-technical problems had a continuous effect in the general transition process. The

personnel who used the legacy tools and the processes did not want this change from the very early stages of this transition. However, most of the time, this resistance has resulted from the personal syndrome the groups that tend to categorically reject new developments tried to generalize. We have seen that to convince the project teams to adopt the ALM 1.0 platform, it is very important to approach the right key people. When they were convinced, they became critical stakeholders in the project.

The tool has been adapted even before going live as user requests and new features have been provided by the team with the help of the practices in the pilot project. To meet the needs of the various departments required a comprehensive knowledge about customizations and selected IBM JAZZ technology. In order to provide direct support when it is essential, it is vital to keep the tool dealer within the reach.

The transition process has been investigated from a high-level business point of view by the senior management. The voluntary and pilot projects have proved that adopting ALM 1.0 is beneficial. The board of the Directors have been convinced by the obtained metric values of these projects and they have decided to adopt the ALM 1.0. Hence, the previous resistance and the criticism have been significantly decreased.

Besides the many benefits this adoption has to offer, there are still some drawbacks that need to be taken care of. Cross platform compatibility is one of them and it needs improvement.In order to cope with new concerns, it is deemed necessary to continue research activities.

4.4. Second Action Research Cycle- ALM 2.0

After ensuring full traceability and transition of requirement, test (manual), error management processes of AKGÜN SOFTWARE to ALM platform within the scope of ALM 1.0, we have started to work on ALM 2.0 to automate the test processes, which are carried out manually, and enable them to operate in an integrated manner with IBM JAZZ and alternative solutions and platforms to serve this purpose will be developed.

The second stage of this study is to automate the testing process. In this way the aim is to achieve an ALM process integrated with the automation of all processes and a more efficient and automated process.

4.4.1. Diagnosis

If the product and product components in the projects are inaccurate, problems such as customer dissatisfaction, increases in the effort, cost and revisions and failure of the projects may occur. Verification and validation activities in the projects are carried out to eliminate or minimize these results.

Verification is to test and secure that the selected work product complies with the defined standards. In the verification activity, work products are tested for functionality regardless of the environment. The verification method must be selected according to the work product.

Validation is to show that a product or component will work as intended when placed in its intended environment. In the validation activity, the real environment or simulation of the work product is generated and the test is performed in terms of functionality in this environment.

Verification and validation methods can be;

- Tests,
- Reviews,
- Prototypes.

Tests within the scope of verification and validation are carried out within the scope of the following flow.

In the projects that take place throughout the organization, the planning of the tests starts with the revealing of the requirements. During the planning phase of the tests, the subjects with the following scope are determined;

- Determination of the test strategies,
- Determination of the test types,
- Risk assessments of the units to be tested,
- Determining the need for training in the persons to perform the test,
- The need for the tools and equipment required for the tests,
- Need for procurement for the tests,
- How much time will be spent for the tests,
- When the tests will be designed,

- When the test will be performed,
- Determination of test termination criteria.

To be able to carry out an effective diagnosis, the relevant groups and key personnel to perform the works have been identified in the first place.

As a result of the interviews, meetings, and reviews of the projects conducted with key personnel, the situation has been evaluated for the automation of the manual test processes and the benefits to be provided have been discussed in detail. The interview results are explained in the following paragraphs.

The tests are carried out in accordance with the requirements of the relevant work product, the established criteria, the test scenarios and the established environment. Tests can be performed manually or using tools. With ALM 2.0, our aim is to automate the realization of tests.

At AKGÜN SOFTWARE, test scenarios are developed within the tool at the projects that use IBM JAZZ case tool and the tests are performed manually and the test results are entered for each test step in the electronic environment over the tool taking the usage rules defined in "*AKGÜN SOFTWARE IBM JAZZ Test and Error Management Guideline*". Recording of the test results also does not need to be kept on the excel forms; they are kept in the space allocated for test scenarios on IBM JAZZ.

The fact that tests are carried out incrementally during development will allow for early detection of errors and less costly error correction. If rework is required on the work products detected to be erroneous (Requirement Development, Technical Solution, Product Integration), the test activity must also be performed again. Regression test is performed to check whether the corrections have disrupted a feature that is already operational. With the correction of the errors identified as a result of the tests performed, the parts that need to be controlled can be followed in the traceability hierarchy created within the scope of the project. The fact that AKGÜN SOFTWARE has more than 400 customers and 45 different products in the IT sector, has increased the need for the automation of the test processes and transition to a faster and more efficient process by choosing a good platform and the importance of the digitalization of the processes.

Improvements to be achieved at AKGÜN SOFTWARE with the test automation are specified below;

- Activation of the processes to offer better quality products and versions to the customers,
- Early detection of the errors,
- Minimization of human errors,
- Automatic running of the regression tests on a daily basis,
- Assisting the test personnel in detecting errors,
- Minimization of the errors of the personnel by performing the repetitions of the tests carried out automatically with the same simulation by the system,
- As the scenarios carried out with the tool are matched to the requirements, anticipating the risks that might occur with automatic and instant traceability,
- Reduction of rework somehow by automating regression tests (same test is performed once and then repeated by system)
- Quick and 7/24 running of the tests,
- Rapid results in version tests,

4.4.2. Action Planning

Several alternative platforms/tools have been researched to automate the tests processes at AKGÜN SOFTWARE. One of the most important criteria in choosing a tool is to be compatible with our CMMI processes and to ensure traceability between projects and processes. On the other hand, adaptation to our projects, in which many different technologies are being used, is also important and the projects/products have been examined in this context and the alignment with these different technologies has been taken into consideration in the research process.

The requirements have been determined in the first place to select the tools and create the best tool set, the best possible alternatives have been identified, and then interviews with appropriate suppliers and POC studies have been carried out.

A committee has been established within the company to carry out the action planning.

First of all, expectations from the tool, namely user needs, have been identified. The user needs identified will also be an input to the tool suppliers assessment and technical specifications of contracts with suppliers.

The committee established have completed the below stages in the action planning phase.

Analysis: At this stage, user requirements and tools have been analyzed in the first place.

Assessment: The assessment has been made according to the criteria determined by the assessment committee for the tools-materials researched. In the assessment stage, the scoring system has been applied and the selection of the alternative, which has the highest score, is planned.

Operation: At this stage, the assessment of the suppliers and the selection of the most appropriate and fulfilling alternative and progress in the necessary studies will be ensured.

During the action planning phase, it has been found out that the teams have the necessary motivation to transit to the test automation tool. It is seen that successful studies carried out within the scope of ALM 1.0 also have an effect in this motivation and this has been expressed in a similar way in similar meetings.

4.4.3. Action Taking

4.4.3.1. Analysis

User needs have been developed within the scope of the test automation tools researched. During the development of user needs, attention has been paid to include the criteria of the Test Process, which are compliant with our CMMI Level 3 processes. The user requirements are detailed in the table below;

Table 14: Test Process and Test Automation Tool User Requirements

1.	Test scenarios should be able to be created.
2.	The review and approval stages of the test scenarios prepared should be managed through the system. Unapproved test scenarios should not be run.
3.	Each test scenario must have a unique number.
4.	Test scenarios should be grouped in n number (Example: Scenarios such as patient registration, update, delete should be created under the Patient Acceptance Module)
5.	The respective test steps of each test scenario should be able to be sequential and numbered.

6.	The expected output for each test step, i.e. the results that must be obtained as a result of performing the test step, should be written and verified.
7.	The test steps of the scenarios to be tested should be able to be created before starting the test (without waiting for the coding to finish).
8.	The test steps should be matched with the requirements developed on the IBM JAZZ Requirement Management tool.
9.	After the matching process, it should be possible to trace which requirement has erred.
10.	In the process of preparing any test scenario, the relevant test scenario may be dependent on other test scenarios. Therefore if there is a pre-scenario condition in the assumptions and it is a previously written scenario, there should be no need to rewrite it (Example: Not every time a logging into the system scenarios is written, only a link can be established with the written object or the previously written scenario should be copied if it will not affect the performance).
11.	Actors in the test tool (user, any tool name or user groups) should be defined.
12.	Users to have prepared and tested the test scenario and time of preparation and testing should be kept in the system and be accessible.
13.	The test steps should be run in the order automatically defined by the system.
14.	Scenarios to be run by the system should be able to be selected and operated independently by the user. These processes should be performed on the basis of any scenario and steps. (Example: Run the scenarios no. 1, 5, 7, 8 in turn or run the steps no.1, 2, 3 of the scenario no.1 or scenario no.5).
15.	The data entered in any test step should be recorded in the system as a data table and associated with the test steps.
16.	If more than one value is entered in the data table, it should be run repeatedly on this step for all alternative values.
17.	The expected output for each record or test data in the data table should be identifiable.
18.	If the values entered are the correct values that the system will accept, it should result in success and continue to the next scenario or step.
19.	The actual output for each step as a result of the execution of the test step (whether or not the result can be explained or shown) should be kept in the system, be in the relevant step and be accessible.
20.	Any test step should be able to display 2 different result states as successful and unsuccessful.
21.	The error levels should be defined and assigned for each test step.
22.	If the test result fails at the relevant step, it should be able to keep the errors in 5 levels (Critical, Very Serious, Serious, Intermediate, Minor), and should be shown in the test results and reports depending on the relevant step.
23.	Test errors for each test step of the system should be automatically assigned by the system. For example, automatically assigning the Serious level to the error message "*Hata*" in red color.
24.	The error levels defined in the system should be assigned priority by the system in the relevant test execution step, but the test specialist should be able to update this error level if needed. However, this process should be specific to the respective test step not to the assigned error level update by the system.
25.	Errors as a result of the test should be in the report format or created by the system automatically. This information includes the following; * Report generation date (date when the report is generated by the user over the system)

	1- Test scenario no and number 2- Step number 3- Error level 4- Explanation and screenshot of the error
26.	Test run times should be calculated and recorded based on grouped scenarios, scenario, and step. Reports and results should be displayed on the relevant record.
27.	The below statistics should be received from the system in the form of report or graphic displays upon the finalization of the tests; * Report generation date (date when the report is generated by the user over the system) * Number of Total Scenarios * Number of Total Steps * Number of Total Successfully Completed Steps * Number of Critical Errors * Number of Very Serious Errors * Number of Serious Errors * Number of Intermediate Errors * Number of Minor Errors
28.	Test scenarios should be created and run for Performance and Integration test types.
29.	Automatic starting of the tests should be performed with the task timer. Example: (every day or on certain days at 19:00 run the scenarios no. 1,4,7,8,9,10 respectively)
30.	For all test results initiated with the task timer, statistical results and graphical reports including date, time, working time, test result states, error levels should be retrievable.
31.	All test results initiated by the task timer should be sent by mail to specific users automatically.
32.	Additions, updates and deletions to the test scenario and steps should be performed on the relevant step or scenario and these procedures should be carried out rapidly and dynamically.
33.	Additions, updates and deletions to be performed in the test scenario and steps should be logged on the revision history of the relevant update and should be listed on the basis of scenario, step and user. These logs should be accessible when required.
34.	It should be ensured that each execution record is archived and can be viewed when requested.
35.	These changes should not be reflected in the archived versions if the steps or scenarios in the archived records have changed afterwards.
36.	The test scenarios statuses (Draft, Testing, Ready for Testing, Error Correction Pending, Completed and Suspended) should be identifiable and the "*Suspended*" status and other manual statuses should be manually assigned by the system. (Example: Unless reviewed and approved, "*Draft*"; if approved "*Ready for Testing*"; if the test is run "*Testing*"; in case of erroneous result "*Error Correction Pending*"; in case of a successful result "*Completed*" should be assigned.
37.	It should be applicable to web-based software projects (HTML, Java, JS, .NET, etc.)
38.	The test tool should be able to record. It is the process of logging into the system as an end user and implementing the scenario and recording of it by the test tool. Test steps are generated automatically.
39.	All kinds of information (steps, placeholder methods, addresses, commands, etc.) on the auto-generated test steps should be manually and easily changed by the user.

40.	In the test tool, test steps should be created manually and automatically. In addition, it should be possible to run the tests by these steps or by means of different algorithms enabling to writing scripts in generic.
41.	The placeholder methods assigned by the system on the steps during the automatic registration process should be reliable or prioritization should be made to ensure that the correct placeholders are used in the correct step.
42.	Test runs should have levels such as fast, medium or slow.
43.	For a 30,000-step project, the duration of automatic test and completion should take no longer than 8 hours (Parallel operation of scenarios without prerequisites or conditions is preferable to shorten the time).
44.	Unit tests should be run and function as integrated to the projects running on such frameworks as java, js, .Net etc..
45.	Training should be provided on the use of the Test Tool and practical training with sample project and sharing the best practices.
46.	References
47.	Frequency of use
48.	Technical and technological support for the problems
49.	Program life and terms of guarantee
50.	Low RAM and CPU Usage

4.4.3.2. Assessment and Acquisition

Alternative testing automation tools that can meet this scope have been investigated. Within the scope of the alternative tools researched, detailed information on the test automation tools examined by installing demo and trial versions on the PCs of the technical assessment committee or Company servers are given below.

Alternative Test Automation Tools

A detailed study has been carried out for 6 months within the scope of test automation. Within this 6-month period, the most common 13 test tools out of many test automation tools in the market have been reviewed. Details of the results of this review are as follows;

4.4.3.2.1. Selenium Web Driver

Due to the fact that selenium has been the most widely used test tool even before the Technical Assessment Committee starts research on the test automation tools, selenium tool has been known and therefore the review has started with the Selenium. The installation of the tool has been completed according to the requirements of the operation of Selenium with Java projects. It has been applied on the test scenarios of a module among our Java projects. Errors have been encountered during the tests, which lasted for approximately 4 weeks, and a number of researches have been conducted on the internet to solve them and a lot of guidelines and forums about Selenium have been used and practical solutions have been generated. Generally, the biggest problem in test automation is the "*id*" and "*location definition*" of the objects. Selenium offers us the option of minimizing these problems to zero with the help of "*xpath*". In addition, the RAM and processor in the system are not forced and it provides ease of use as it is simple and plain. As we do our automation with codes, it provides rapid solution and reduces loss of effort.

We used the Selenium IDE and firebug plugin for record operation and object definitions. In summary, it is concluded that Selenium meets our expectations to a great extent as it is free and provides conveniences.

Table 15: Selenium Test Tool Advantages and Disadvantages

Advantages	Disadvantages
1 – Free 2 – The most widely used test automation 3 – Integration with Java and C# languages 4 – Supports regression tests 5 – Provides rapid feedback 6 – Ensures that test scenarios are repeated 7 – Easy to use 8 – Integration with such tools as IBM Jazz 9 – Ease of error management 10 – Low RAM and CPU use 11 – Capable of integration with TestNG and reporting 12 – Ease of installation	1 –Support only web-based applications 2 – Weak in terms of support

4.4.3.2.2. Hp QTP – UFT

HP's testing tools are known to all companies and two testing automation tools from HP have been reviewed by our Technical Assessment Committee. One of these is LoadRunner, the other is QTP-UFT. QTP is alleged to be the second most used test tool after selenium. It is one of the programs with a wide usage.

Table 16: HP QTP-UFT Test Tool Advantages and Disadvantages

Advantages	Disadvantages
1 – Ease of writing the Test Cases 2 – Record and Playback capabilities and arrangements 3 – Successful object definition 4 – Supports such languages as C# and Java 5 – Ease of use 6 –Sustainable test iterations 7 – Successful analysis and reporting system	1 – It is paid 2 – Installation is hard and challenging 3 – High RAM and CPU use

4.4.3.2.3. Hp LoadRunner

This tool is the ideal tool for performance testing. This licensed product is available free of charge up to 50 virtual users. There are components within that cover many platforms. As the examinations by the Technical Assessment Committee are made within the scope of our projects using web technologies, "*Truclient*" component has been used. When the program is opened, it gives a high quality and safe feeling and when we start recording for the tests, it is seen that the truclient works in conjunction with the browser and the tests are started. Although the program is very attractive for small projects, planning should be made very well in complex projects with many components otherwise the system can be forced and complexity may increase. The most advantageous position is the object definitions. The program has its own locater. It also makes efficient xpath definitions. In summary, even though LoadRunner meets most of our expectations, it will not be sufficient for AKGÜN SOFTWARE due to system performance, complexity and lack of code access mechanism.

Table 17: HP LoadRunner Test Tool Advantages and Disadvantages

Advantages	Disadvantages
1 – Ease of writing the Test Cases 2 – Record and Playback capabilities and arrangements 3 – Successful object definition 4 –No code information 5 – Ease of use 6 – Successful analysis and reporting system	1 – It is paid 2 – High RAM and CPU use 3 – Difficulty in complex and large structures

4.4.3.2.4. Sencha Test Studio

Sencha Company has announced its name with two test tools. These are Siesta and Test Studio. The Test Studio program was firstly examined by our Technical Assessment Committee and the extjs codes within the scope of our projects using Java have been integrated in the tool during installation. It is concluded that the program can be beneficial bot with its interface and ease of use however as the program is in beta version, its failure to record and deficiencies in the system have shortened our time for review and assessment. In summary, although it does not provide solutions in the short term with the processes and projects of AKGÜN SOFTWARE, it is a compatible program that can be considered in the long term.

Table 18: Sencha Test Studio Test Tool Advantages and Disadvantages

Advantages	Disadvantages
1 – Ease of use 2 –Access to codes 3 - Record and Playback capabilities and arrangements 4 – Low RAM and CPU use 5 – Successful analysis and reporting system	1 – It is paid 2 – As it is beta version, there are many deficiencies

4.4.3.2.5. Siesta

Siesta is more stable than Test Studio. However, it is not possible to record on the trial version of the product so it can only be examined from specific examples. As a result of research from different sources, the program is simple, useful but not sufficiently comprehensive and compatible with our project and requirements.

Table 19: Siesta Test Tool Advantages and Disadvantages

Advantages	Disadvantages
1 – Capable of running Unit and Functional tests successfully 2 – Access to the codes 3 – Record and Playback capabilities and arrangements 4 - Low RAM and CPU use 5 – Ease of use	1 – It is paid 2 – Sufficient support is not available online

4.4.3.2.6. Microsoft Test Manager

One of the companies that come to mind first when buying a product is Microsoft. Test tools of Microsoft have been examined by our Technical Assessment Committee. Test Manager is generally used extensively for manual tests and works compatibly with C# language for code side. Therefore, it is quality product but not compatible with our projects.

Table 20: Microsoft Test Manager Test Tool Advantages and Disadvantages

Advantages	Disadvantages
1 – Quality of Microsoft technologies 2 – Has a nice interface 3 – Ease of use 4 – Interaction with the Visual Studio 5 –Ideal for manual tests	1 – Only supports manual tests 2 –It is paid 3 – Installation is hard and challenging

4.4.3.2.7. Oracle Test Suite

Since our Technical Assessment
Committee uses AKGÜN SOFTWARE's Oracle programs, Oracle Test Suite tool has been taken into consideration but problems such as not providing browser support have been encountered and it is concluded that it does not comply with our projects.

Table 21: Oracle Test Suite Test Tool Advantages and Disadvantages

Advantages	Disadvantages
1 – Capable of functional and loading tests 2 – Shortens development time 3 – Security of the data during the test 4 – Integration to Script platform	1 – It is paid 2 – Supports only web-based applications 3 – No longer supports Java 4 – Not having a wide area of use

4.4.3.2.8. Silk Performer

The Technical Assessment Committee examined the Silk Performer, a test tool from Borland firm. The program is easy to install but the tool only includes performance tests and does not fulfill our criteria therefore our examinations have been terminated.

Table 22: Silk Performer Test Tool Advantages and Disadvantages

Advantages	Disadvantages
1 – Capable of running load and stress tests 2 – Successfully runs performance tests	1 – Difficulty in use 2 –Has no timing 3 – Runs only performance tests

4.4.3.2.9. Sahi

Although it is not widely used in the testing automation sector, it has been seen to be added to the list for review by our Technical Assessment Committee. It is a paid tool and it offers free use for 1 month. The program works on the console and integrates with the browser. Although it has many useful components, it is a highly complex tool. Various problems in terms of difficulty in use, test merge and

proxy have occurred. In short, although it is a paid tool, it has a very complex structure and may cause integration problems in our projects.

Table 23: Sahi Test Tool Advantage and Disadvantages

Advantages	Disadvantages
1 – It is web-based 2 – Integration with the scripts 3 – It has a recording system 4 – It is capable of object definition	1 – Not has a sufficiently wide use 2 – Ideal for performance tests 3 – Insufficient in terms of interface

4.4.3.2.10. Selenium IDE

Selenium has been assessed separately from Web Driver and attracts attention as it is free and widely used, is rapid and easy to use. This program is usually very useful in non- complex projects. However, as complexity increases, accessibility and error detection takes a longer time and it is necessary to spend a lot of time to correct. The locator feature is inadequate in some cases, and the dynamic id is troublesome in the software used. In summary, it is insufficient for long and complex projects.

Table 24: Selenium IDE Test Tool Advantages and Disadvantages

Advantages	Disadvantages
1 – It is browser based 2 – Timer 3 – Speed 4 – Low RAM and CPU use 5 – It is free 6 - Save & record processes are easy	1 – Object definition problems 2 – File order complexity 3 – Solution difficulty in case of problems as the code is not supported 4 – Interface and processes are not user-friendly

4.4.3.2.11. Apache Jmeter

Apache Jmeter is open source software designed to measure performance. It is used to test web applications. The examinations have not been elaborated and have been discontinued as it is identified that user requirements have not been fulfilled in the first stage.

Table 25: Apache Jmeter Test Tool Advantages and Disadvantages

Advantages	Disadvantages
1 – Ease of use 2 – Code integration 3 – It is free	1 – Installation difficulty 2 – Insufficient support 3 – Not commonly used

4.4.3.2.12. SoapUI

SoapUI tool has been examined by our Technical Assessment Committee and it is seen that Soap web services are tested. It is not possible to use it in our projects because it only performs web service tests and does not fulfill the user requirements.

Table 26: SoapUI Test Tool Advantages and Disadvantages

Advantages	Disadvantages
1 – XML based operation 2 – Tests web services	1- Not runs functional and load test

4.4.3.2.13. IBM Rational Test

IBM Rational Test tool has been examined by our Technical Assessment Committee and the first impression is that it has complex interfaces to use.

Table 27: IBM Rational Test Tool Advantages and Disadvantages

Advantages	Disadvantages
1- Successful object definition 2- Ease of IBM JAZZ integration 3- Ease of writing test scenarious 4- Successful analysis and reporting system and suit our processes 5- Applicable for desktop, mobile, web based projects 6- Record and playback capabilities and arrangements	1- High resource usage (RAM, CPU, Disk) 2- It is paid 3- The performance test tool is also charged extra.

The comparison table for the tools examined above by the Technical Assessment Committee, which includes the technological compatibility of the tools with the projects of different programming languages of AKGÜN SOFTWARE, is as follows;

Table 28: Technologies Used in the Project and Functional Test Tool Compatibility

Tools / Projects	IBM Rational Test Workbench	HP Unified Functional Testing	SELENIUM	MICROSOFT	ORACLE TEST SUITE	RIATEST	SAHI	SENCHA	TELERIK
P1	Yes	Yes	Yes	Yes	No	Yes	Yes	Yes	Yes
P2	No	Yes	No	No	No	No	No	No	No
P3	No	Yes	No	No	No	No	No	No	No
P4	Yes	Yes	No	No	No	No	No	No	Yes
P5	Yes	Yes	No	No	No	No	No	No	Yes
P6	Yes	Yes	Yes	Yes	No	Yes	Yes	Yes	Yes
P7	No	Yes	No	No	No	No	No	No	No
P8	No	Yes	No	No	No	No	No	No	No
P9	Yes	Yes	Yes	Yes	No	Yes	Yes	Yes	Yes
P10	Yes	Yes	Yes	Yes	No	Yes	Yes	Yes	Yes

As a common disadvantage we have evaluated within the scope of test automation tools; even though we are planning a tool transition to automate the preparation and running of test scenarios and the fully integrated ALM automated process, we are unable to position the tools in all processes/stages of manual tests. Because ad-hoc tests have an important place in the software products, Ad Hoc tests, which are made as if were in the customer's place, may have significant errors and this type of test cannot be done by means of the tool due to its nature.

Each one of the test automation tools includes some aspects that must be resolved and/or overcome and this may be listed as follows;

Table 29: Comparison Table for the Problems and Risks of the Alternative Test Automation Tools

TOOLS	PROBLEMS	RISKS
IBM RATIONAL TEST WORKBENCH	• During the test scenario preparation process « Record », all interfaces of the screen are downloaded and therefore each test step slows down and waiting time is 5 seconds instead of 1 second. • There is not much information on the Internet for technical problems and solutions. • The client and update installation and the configuration to come afterwards are carried out by the company and therefore there is a waiting time. Additional time during notification to the company and return from the company.	• Problems cannot be solved because there are not enough resources about the problems encountered in the use of products on the Internet and their solutions.
HP Unified Functional Testing	• No problem identified.	-
SELENIUM	• Desktop-based applications not supported • Lack of competence in mobile applications • Project-based software solutions are required because object locator capabilities are not developed for the latest technologies (ext js, etc.)	• Some problems may arise in scenario recording and execution (freezing, failure to find the objects, passing the related step, etc.) • There is no support when the problem is encountered (solution can only be found by researching the resources on the internet)

TOOLS	PROBLEMS	RISKS
MICROSOFT	• Desktop-based applications not supported • Lack of competence in mobile applications • Project-based software solutions are required because object locator capabilities are not developed for the latest technologies (ext js, etc) • There is not much information on the Internet for technical problems and solutions.	• Since there are not enough resources about the problems encountered in the use of products on the Internet, solutions cannot be produced in all cases.
ORACLE TEST SUITE	• Desktop-based applications not supported • Mobile applications not supported • Oracle browser support has been terminated. The browsers supported are very old versions and do not comply with our projects. • There is not much information on the Internet for technical problems and solutions.	• Since there are not enough resources about the problems encountered in the use of products on the Internet, solutions cannot be produced in all cases. • As there is no browser support, test tool cannot be used.
RIATEST	• Desktop-based applications not supported • Mobile applications not supported • The firm does not provide support in Turkey • There is not much information on the Internet for technical problems and solutions.	• Since there are not enough resources about the problems encountered in the use of products on the Internet, solutions cannot be produced in all cases.
SAHI	• Desktop-based applications not supported • Mobile applications not supported • The firm does not provide support in Turkey	• Since there are not enough resources about the problems encountered in the use of products on the Internet,

TOOLS	PROBLEMS	RISKS
	• As there is no interface, it increases complexity	solutions cannot be produced in all cases.
SENCHA	• Desktop-based applications not supported • Mobile applications not supported • The firm does not provide support in Turkey • As the test tool is in beta version, there are many deficiencies.	• Since there are not enough resources about the problems encountered in the use of products on the Internet, solutions cannot be produced in all cases.
TELERIK	• The firm does not provide support in Turkey • As the test tool is in beta version, there are many deficiencies. • As the Record feature is limited and not compliant with the web-based projects, its ability to define objects is limited and cannot be changed.	• Since there are not enough resources about the problems encountered in the use of products on the Internet, solutions cannot be produced in all cases.

Scoring Alternatives and Evaluation Systematic:

In our organization, decision analysis is made at the stage of evaluation of alternatives for each decision.

Decision-making / selection criteria are determined during the evaluation of alternatives. The appropriate selection criteria and evaluation methods are used to select the appropriate alternatives. At least 3 alternatives are determined. Criteria; plans and targets compliance, quality expectancy, customer satisfaction, delivery time, warranty period, training support, technical criteria, user requirements, technological compatibility etc. criteria and considerations are formed.

The selection criteria are rated as 1 lowest and 5 highest by order of importance. The stage of rating the selection criteria is determined by the common opinion of the people, who form the alternatives, and the decision makers. The current situation and conditions, in which the subject to be decided is, are also taken into consideration. At

the stage of determining the selection criteria, whether the criteria are included in the evaluation or not and the reasons are also evaluated. In this case, a check mark is added next to the criteria included in the evaluation, and why it is selected is indicated. A description of why the criteria are not included in the evaluation is also indicated.

The determined alternatives are taken into evaluation according to each selection criteria that is formed. The person, who formed the alternatives, may refer to the scoring systematic, old experiences, comparison with similar ones serving the same purpose (survey and research results, promotional presentations), cost analysis techniques, prototypes, risk analysis, business opportunities assessments and tests among the alternative evaluation methods.

Selection Criteria and Scoring Method is commonly used as the evaluation method in evaluation and the alternative with the highest score is selected. If there are other evaluation methods in besides the scoring systematic, details are explained.

When determining the selection criteria and evaluating alternatives, the level importance of the criteria should be taken into consideration. 5 is the highest level of importance, whereas 1 is the lowest level of importance. The scoring is made between 1-5.

Then, alternative's score of meeting the criteria is scored again according to the level of importance given for each criteria. The level of importance of the criteria and the alternative's score of meeting the criteria are multiplied and the score for that criteria for the alternative is obtained. This method is applied to all determined criteria and the scores are summed. The alternative with the highest score is presented as a suggestion. Within the scope of the approved alternative, the required supplier and contract management and other administrative processes are started and technical works begin.

Tablo 30: Rating the Alternative's Score for Meeting Criteria and Descriptions

Score	Level of Meeting Criteria	Description
1	Inadequate	The alternative does not meet this criteria.
2	Low	The alternative meets this criteria on a low level.
3	Average	The alternative meets this criteria on an average level.
4	High	The alternative meets this criteria on a high level.
5	Very High	The alternative meets this criteria completely.

The criteria for assessing the alternative test tools are as follows. The comparison is made based on the criteria identified within the scope of each alternative test tool.

Comparison criteria of the tools under IBM, Selenium, HP, Sencha, Oracle, Silk Performer, Microsoft, Siesta, Sahi, Jmeter, SoapUI have been examined.

Table 31: Test Automation Tool Assessment Criteria and Importance Levels of the Criteria

Criteria	**Importance Level**
IBM JAZZ integration	Very Serious
Useable in web projects	Critical
Capable of associating and prioritizing the use statuses and steps and requirements for the steps	Critical
Recording and execution of the Test Steps with end-user behaviors	Intermediate
Ability to carry out addressing and execution coding manually	Critical
Ability to carry out the test steps on the basis of project, module, scenario and step, automatically and by means of task scheduling	Critical
Speed and time performance comparison for execution	Critical
Output for test result report	Critical
Automatic reporting of test results	Intermediate
Reporting of the results to the officials through tools such as SMS and mail	Slight
License and Costs	Strategic
Sustainable and expandable feature	Critical
Functional testing feature	Critical
Unit Test feature	Strategic

Criteria	Importance Level
Integration Test Feature	Strategic
Performance Test Feature	Strategic
Accessibility and Ease of Use	Serious
Backup and Logging	Very Serious
Integration with the project	Strategic
Training Support	Critical

Importance level explanations of the criteria are given below;

Table 32: Test Automation Tool Assessment Criteria Importance Level Explanation

Importance Level	Explanation of the Importance Level
Critical	Processes and technically mandatory.
Very Serious	It is not sufficient that it complies with the processes, it is expected that inputs and outputs have and measure a user-independent relation with the processes.
Serious	Most of the current effort can be reduced.
Intermediate	Little of the current effort can be reduced.
Slight	Extra features
Strategic	Its strategic importance is identified according to the institutional goals.

As a result of the researches and evaluations conducted, it has been decided to conduct more detailed researches on the IBM, HP and Selenium tools with the highest rate of fulfilling the user needs in 13 test automation tools and to advance the next stage evaluations through these 3 alternatives.

As a result of the demonstration and installation procedures, the estimated time and effort for the transition of the test scenarios and test processes in the existing projects to 3 alternative tools (HP, IBM, and Selenium), which are included in the final evaluation list, have been calculated. Among the projects, the total transition efforts

(man*day) of the 10 projects with the most common customer portfolio, for each tool are as follows;

Table 33: Automation Transition Times According to the Test Automation Tool Alternatives

FOR 10 PROJECTS	Test Scenario Numbers	Test Step Numbers	IBM (man*day)	Selenium (man*day)	HP (man*day)
TOTAL	**4.544**	**15.211**	**91,62**	**67,77**	**37,69**

Effort calculation items that serve the input for the above values under each tool are as follows and the additional efforts to occur additionally in the transition of the test scenarios to the automation in the continuation of the installation are given below;

Table 34: Test Automation Tools Transition Effort Calculation Method

TOOL	ADDITIONAL EFFORT ASSESSMENT
IBM RATIONAL TEST WORKBENCH	Combining, arranging, moving and updating multiple scenarios creates effort. 1 unit of work has been measured in about 2 units. Estimated effort for the combination of 20-step scenario with another 10-step scenario is 0.1 man * days (combination, transfer, update).
HP Unified Functional Testing	No additional effort is required.
SELENIUM	• Object locating (For each scenario 0,5 man*day) • Data table (For the project 1 man*day) • Ext JS Compatibility (For project 20 man*day) • Reporting (For all projects, 7 man*day) • Verification (For each scenario 0,1 man*day)

As a result of the assessments, 3 alternative tools (HP, IBM and Selenium tools) have been identified to fulfill the criteria to a higher degree and the alternatives, which are in the final list, have been evaluated according to the following criteria. As a result of the scoring over 100, the results are as follows;

Table 35: Test Automation Tool Alternatives (3 Tools on the Final List) Assessment and Scoring Criteria

Criteria
"*Record*" and "*Play*" Feature
Desktop application support
Browser Support
Supported Languages
Operating System Support
Test Management Tool Integration
IBM Jazz Integration
Test Development Environment
UI Object Identification
Mobile Support
Image-based Testing
Reporting Capability
Complexity
Script Development Time
Data Usage (CPU + RAM)
Product Support
Test Debugging Support
Automatic "*Exception Handling*"
Popup Handling Support
Object Parameterization
External Library Integration
Object Oriented Script Support
Ease of Test Case
Object Storage
Multiple Parameter Identification
Storage of Data
Accuracy
Productivity
Cost
Parallel Operation
Ease of Use

Table 36: Assessment Results According to Test Automation Tool Alternatives (3 Tools on the Final List) Scoring Method

HP QTP	SELENIUM	IBM
95,16667	**68**	**82,16667**

On the other hand, according to the 45 user requirements identified in Table 11, the total result values are as follows when the last 3 alternatives are scored for fulfillment level of the requirements over 100;

Table 37: Comparison Scores of the 3 Test Tool Alternatives on the Final List for Fulfillment of the User Requirements

IBM	HP	SELENIUM
82,54385965	83,75439	73,245614

As a result of the evaluations made as seen above; it has been decided to select HP test automation tool, which got the highest score in both meeting the user requirements and fulfillment of the assessment criteria. The results have been reported to the supplier after the selection has been made and our formal supplier contract management processes have progressed.

4.4.3.3. Base Installation, Customization and Development and Adoption of Pilot Projects

The installation process of the test tool during the installation phase has been carried out on the space allocated on the server. Installations in this space have been completed as a result of the joint effort of the technical teams appointed within the Company and the HP officials, and LDAP integration has been commissioned by transferring the users to the system and by means of license assignments.

As the pilot implementation, project made up of 82 modules and 35.000 test scenarios and with a team of 30 engineers and using the Java and Javascript programming languages has been elected, the project's first 15 critical modules and version test scenarios have been started to be transferred to the system.

4.4.3.4. Going Live

HP UFT is used as a test automation tool during the testing phase of the software lifecycle. Processes such as reporting the test scenarios, related steps, test data and test results of software tests to be performed within the scope of the project are

carried out. Tests in AKGÜN SOFTWARE are performed automatically on HP UFT by comparing test scenarios in IBM JAZZ to the test scenarios in HP UFT.

The test scenarios of the P1 Project are actively kept and GUI Tests are performed. In this context, Critical scenarios (scenarios required for the operation of the system) and version tests are performed by HP UFT.

4.4.4. Evaluating

In Table 35 below, the total number of scenarios and number of steps transferred to the test automation tool within the scope of HP Test Automation transition are shown.

Table 38: Data Table for the Pilot Project Used for Transition to HP Test Automation Tool

Within the Scope of P1	Value
Total Number of Scenarios Transferred to the Test Automation Tool	923
Total Number of Steps Transferred to the Test Automation Tool	3.692
Time to Transfer a Test Scenario to Test Automation (for a Scenario with an average of 10 steps)	30 minutes

Table 36 shows the metrics before and after transition to the HP test automation tool. AKGÜN SOFTWARE aimed to distribute a new version to the field every 2 weeks as a result of the changes and additional requests made by the customers within each product, but this process could not be realized on a continuous basis before the transition to the test automation and deviations occurred in these times, because the manual tests were causing the process to progress slowly. However, in the test automation tool, it is possible to run the tests by defining the desired number of repetitions and the desired time period. The average number of additional requests sent on a daily basis by the customers that are transferred to IBM JAZZ and HP test automation and use P1 product is 38. These are developed and included in the new version and distributed to the customer. The time required for the version tests and the duration of the resource and distribution have decreased considerably.

Table 39: Before and After Comparison Table for HP Test Automation Transition

Metrics	Before the Test Tool	After the Test Tool
Version Test Time	7-8 hours	2-3 hours
Number of Persons Required for Version Tests	4	1
Version Release Time	20 days	14 days
Test Time for the Test Scenarios (for a Scenario with an average of 10 steps)	10 minutes	3 minutes

4.4.5. Specifying the Learning

As a result of the fact that the advantages and process efficiency of ALM 1.0 are seen by the teams, the teams seem much more willing and enthusiastic for ALM 2.0 activities. Both the management support and the investments made have had a positive impact on the team and it is seen that they make the sacrifice for the integrated management of the processes and take more part in the detail works compared to ALM 1.0 and other teams support the teams in their works, too.

In the stage of transition to the test automation tool, many technical lessons learnt related to the use of the tool and the transition to the test automation tool have been obtained and they can be listed as follows;

- During or after the installation, attention should be paid to add the add-ins from the setting screen, otherwise problems such as inability to carry out the test may be encountered.
- It is advantageous to have a detailed analysis of the test scenarios to be transferred to the automation tool before the work and to transfer the similar scenarios/interrelated scenarios at the same time for the more efficient use of time and performance. For this reason, all scenarios should be separated in modules and their interconnections and the business plan should be analyzed and detailed.
- Names of the scenarios and the name of the object repository should be meaningful during the creation of the tests. For example; it is important that when returned to the scenario after a month or another user sees the scenario, the names should be predictable.

- To ensure that frequently used scenarios are not written again and again and are easy to respond, it is proposed to create Function. For example; such processes as log-in for all of the scenarios or the addition of diagnostics for most of them are required. Reproducing these processes in each scenario creates a waste of time.
- The use of the IF-ELSE logic for each line is useful for detecting the exact error, exact location of the error during the creation and execution of the test scenarios.
- At the end of the scenarios created, it is necessary to close the program and move to a new scenario. The use of the "*browserQuit*" code at the end of the scenario saves time.
- If a written scenario contains common areas with a scenario in a different module, the same places need to be written over and over again. Instead of this, one can save time by doing "*Call to copy of Action*" instead of typing common areas all over again.
- If necessary, it is important to back up the test by export.

4.5. Third Action Research Cycle- ALM 3.0

4.5.1. Diagnosis

One of the most important steps in ALM is the integrated and fully integrated study of the processes. With the inclusion of IBM JAZZ and HP UFT tools in the software lifecycle, the need to review of a number of integration points and the interoperability of these tools with other tools used at AKGÜN SOFTWARE has arisen. This review is important in terms of evaluating both our institutional systems and the contribution of investment to ALM processes.

As a result of a series of meetings, interviews and examinations with related Managers and teams within the company; in this study, it has been concluded that after the transition of the tests manually run on IBM JAZZ tool to the HP Test Automation Tool, the interoperability of these two tools in the Test Processes should be evaluated and the integration points should be identified and integrated study should be reevaluated.

As a result of the evaluations made; the following problems and improvements have been identified;

- The test scenarios and steps created in IBM JAZZ Quality Management should be paired with the test scenarios and steps generated in the HP UFT tool and dual traceability is required. In fact, in the event that any requirement changes, it is not possible to see, which lines of code are related in HP UFT, to ensure the relevance and traceability of the respective test scenarios.
- The necessity of matching the changes/updates to the requirements as a result of the changes in the works within the scope of the Project or change requests received has occurred. In this context, the parts affected in IBM JAZZ by the works assigned to JIRA cannot be tracked.
- It has been evaluated that the automatically opening the errors occurring as a result of running of the tests as a task in JIRA will be beneficial to increase the efficiency of the processes.

4.5.2. Action Planning

Within the scope of the planning of the actions, focus has been placed on the problems that have been identified in the diagnosis phase and at the meetings.

A committee has been established within the company to carry out the action planning. Within the Committee established; persons competent in IBM JAZZ, HP UFT and JIRA tools, who are authorized to perform all necessary technical interventions, have been assigned. The process of evaluating the possible integrations and commissioning the necessary technical research and related improvements is carried out by these persons.

Analysis: In this context, a detailed analysis of the solution methods of the problems identified in the diagnosis phase and arrangements to be made in the screens and operating and usage scenarios of the tools has been performed.

Assessment: Some of the analyzes and user integration needs identified are resolved by data entry to the fields on the screen and some of them require intervention to code and technical sub- section. As a result of the evaluations and the interviews made with the suppliers and other studies conducted for these tools, the related needs have been considered to be manageable.

Operation: As a result of the examinations carried out, it will be ensured that adaptations to be made within the tools are commissioned. In this context, effort and work planning will be made and commissioned as planned.

Only as a result of a series of meetings with IBM and within the scope of the above-described automatically opening the errors identified as a result of the execution of the tests as tasks in JIRA and additional research, it is seen that IBM JAZZ has no such integration for the time being. The efforts on IBM side continue in this regard and it is concluded that this integration will be made by paying additional fees. Therefore it has been decided to perform this integration, i.e. opening the errors as task to JIRA, manually.

4.5.3. Action Taking

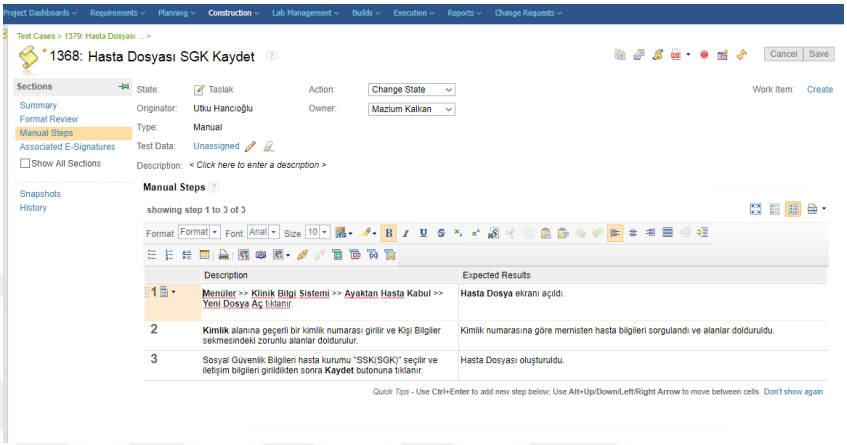
Within the scope of matching the test scenarios and steps created in IBM JAZZ Quality Management with the test scenarios and steps generated in the HP UFT tool and ensuring dual traceability;

In order to integrate the test scenarios created in the IBM JAZZ ALM tool with the HP test automation tool, the steps in the IBM JAZZ scenarios are added to the code lines of the relevant test in the HP test automation.

At the same time, the status of the tests is seen in the IBM JAZZ tool and the test results (successful-unsuccessful) are monitored in the IBM JAZZ tool.

Examples of integrations created on this basis are given below;

In the following screen, the patient registration scenario has 3 step definitions and the expected result.



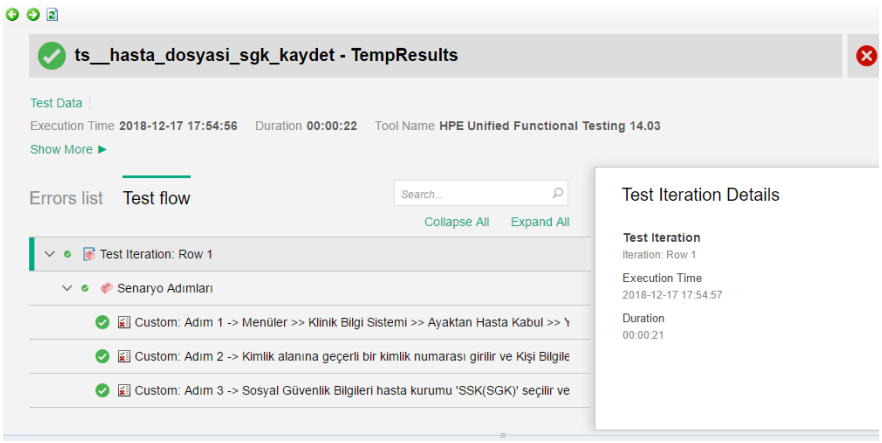
Screen Shot 1: Screen Shot of the Patient Registration Steps in IBM JAZZ

The scenarios entered in IBM Jazz are copied to the HP test tool and displayed in the report. The following screen shows the method of matching the IBM JAZZ test steps in HP UFT.

```
Const step_1 = "Adım 1 -> Menüler >> Klinik Bilgi Sistemi >> Ayaktan Hasta Kabul >> Yeni Dosya Aç tıklanır."
Const step_2 = "Adım 2 -> Kimlik alanına geçerli bir kimlik numarası girilir ve Kişi Bilgiler sekmesindeki zorunlu alanlar doldurulur."
Const step_3 = "Adım 3 -> Sosyal Güvenlik Bilgileri hasta kurumu 'SSK(SGK)' seçilir ve iletişim bilgileri girildikten sonra Kaydet butonuna tıklanır."
```

Screen Shot 2: Screen Shot Showing the Matching of the Patient Registration Scenario in IBM JAZZ to the Scenario and Steps in HP Test Automation Tool

A test scenario result output, whose tests have been completed, can be displayed in the HP test automation tool as follows;

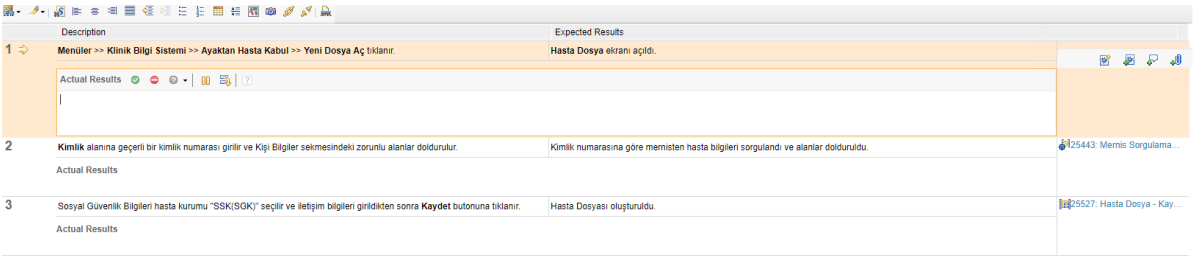


Screen Shot 3: Screen Shot Showing the Status of the Scenario Run in HP Test Automation in IBM JAZZ Screen (Running, Completed etc.)

In the HP test automation tool, a test scenario, where test runs have been completed, can be seen in IBM JAZZ (passed, failed) and an example is shown in the following screen;

Screen Shot 4: Screen Shot Showing the Status of the Scenario Run in HP Test Automation in IBM JAZZ Screen

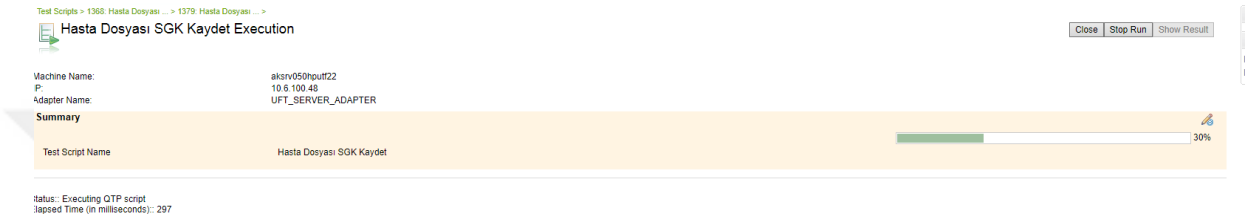
Within the scope of the integration of IBM JAZZ and HP test automation and dual traceability; as shown in the screen shot below, the status of a manually run test scenario, i.e. manual operation status, is shown not HP test automation. In this example, the test result outputs of each test step are processed manually.



Screen Shot 5: Screen Shot of the Information on Manual Running of the Test in IBM JAZZ

In addition, the parts on the left side of the above screen (25443: Mernis Sorgulama...), are the part, where each test step is matched with the requirement, the connection and link to the requirements the test steps relate are given.

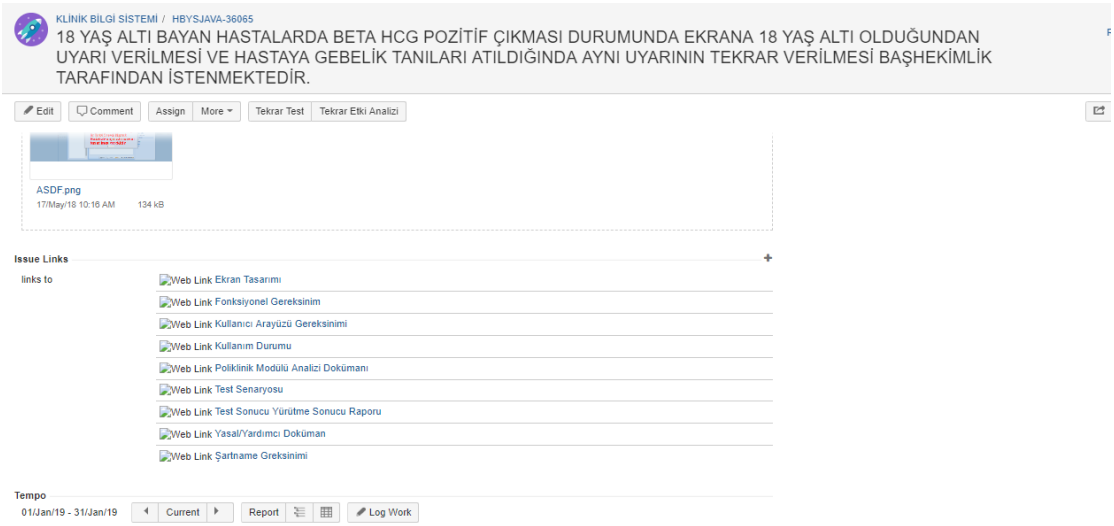
Information regarding the tests executed in the HP test automation tool, as seen in the screen shot of the IBM JAZZ section, such as the test is run with HP, instant progress of the test scenario run, termination of the test, if required, can be seen on IBM JAZZ.



Screen Shot 6: IBM JAZZ Screen Shot Showing That the Test Run with HP is Automatic

The data described in the screens above are generated automatically in the respective areas of the tools as a result of our integration between HP and IBM tools.

The necessity of matching the changes/updates to the requirements as a result of the changes in the works within the scope of the Project or change requests received has occurred. In this context, the parts affected/updated in IBM JAZZ by the works assigned to JIRA cannot be tracked. In this context, the links of the impact analyses of the updated requirements in IBM JAZZ, test scenarios are copied to the description section of the relevant work in JIRA. Thus, traceability is provided through links.



Screen Shot 7: JIRA Screen Shot Including JAZZ Links of the Parts Updated As a Result of Change in IBM JAZZ

4.5.4. Evaluating

As a result of the integration of the solutions created under ALM 1.0 and ALM 2.0 in AKGÜN SOFTWARE, the test scenarios transferred to the HP test automation tool and the test scenarios included in the IBM JAZZ have been matched in whole. In this context, the 923 test scenarios and related lifecycle processes have dual traceability.

Table 40: Test Process Work Product Matching between HP and IBM JAZZ

Within the Scope of the Project1	Value
Total Number of Scenarios Transferred to Test Automation Tool	923
Total Number of Scenarios Matched with IBM JAZZ	923

As of the end of 2018, 2.254 artifacts of the projects transferred to IBM JAZZ in total have been revised in IBM JAZZ. Approximately 10 change requests or new requests from AKGÜN SOFTWARE customers are received on a daily basis and work products are subject to frequent revision/change within this scope. JAZZ access links of the work products that are updated as a result of the change request/additional requests are included in the relevant descriptions in JIRA.

Transition to the test automation tool and IBM JAZZ is considered to require additional efforts and labor force loss during the process of reflecting the received changes to the relevant documentation and keeping it up-to-date. The loss of traceability as a result of the changes may result in missing the parts to be tested and

ultimately lead to internal and external customer dissatisfaction. However, more efficient management of the product and software lifecycle processes is ensured as a result of the transition to test automation and increased traceability.

4.5.5. Specifying the Learning

The fact that the tools are not open to user development during the implementation of inter- tool integrations and that sufficient information is not available to intervene in the system, and also that firms such as IBM make late return to user demands, can delay this process. For the management of these processes, AKGÜN SOFTWARE enjoyed the advantage of having senior developers and making solutions by performing the necessary examinations and researches. It is important and practical for companies to be able to produce solutions that can meet their needs on internal tools.

At the point of ensuring integration, our teams are willing and guide the process by providing opinion and the continuity of the studies to complete the customizations on the tools in a short time will be beneficial in achieving the result.

4.6. Discussion

Within the scope of this study, action research was conducted regarding the application of ALM in the industry and it was observed that there had been no action research that involves the cycles, and essentially the cycles each of which is a separate action research, within this scope and that is comprehensive and involves detailed research and demonstration processes. For an action research study to be successful, firstly, the problem to be solves has to be original and idiosyncratic and also, it is expected to achieve correct and authentic results as a result of a conducted research [55]. The problem that is addressed in this study, the studies that were conducted, and their results have been deemed unique, invaluable, and successful within AKGÜN SOFTWARE.

The results of this research have been discussed in continual meetings with the Consultants and other information Companies.

It has been realized in these meetings held within this scope;

As ALM Tools:

For the management of changes in the requirements, IBM ClearQuest tool is also used besides the requirements management tool. Change management tool is a tool that is used for completing the first versions of software, in particular, in the software development process and following errors and improvement recommendations after their first integrations. As a result of conducted research, it is realized that the Telelogic DOORS Requirements Management tool is used by some companies in Turkey. Some Companies in the sector, on the other hand, use Rational Doors.

As Test Automation Tools:

Today, there are various software testing tools that are used for different purposes. Selenium Webdriver has been known as the most commonly used web automation tool in the market. It is used for conducting functional tests. The main reasons for its widespread use in today's sector:

- Free due to open-source code,
- Supporting a variety of software language,
- Operating with all operating systems,

- Providing coding experience,
- High-performance,
- Framework flexibility

And again in the meetings with other information Companies, it has been observed that HP UFT Test Automation Tool is also commonly used.

Moreover, there are also several information that are commonly used in relation to the non-functional tests and tools. Performance and security tests are the examples of non-functional tests. Open-source code Jmeter and paid HP LoadRunner are the examples of commonly used tools for performance tests in the market. The most commonly used ones in security tests, on the other hand, are IBM AppScan, Webscarap, and BurpSuite tools.

As a result of the examination, it was observed that these four tools are superior to each other in certain areas. Regarding the criteria compared in this study and reading some reviews available on the internet, it can be realized that the HP Quick Test Professional product comes a step forward. In the ranking based on the review, comparison, and scoring on the IT Central Station's website [56], it is seen that HP QTP and IBM RFT are ranked at the top as well. Selenium is also a product of serious use; however, among these four software testing tools, HP QTP and IBM RFT stand out. In terms of cost, HP QTP is not only one of the four software, but also it is one of the tools with highest cost among all testing tools. However, it is preferred due to the low possibility of invisible cost, since the test procedures are easy, the need for software expertise is low, and the training costs are low compared to other tools. Even though the Selenium tool is free, it can be more expensive in periods of 5-10 years due to its complexity and high level of software knowledge. This is due to the higher salary of test staff with high software skills. Regarding all these, the selection of the testing tool should be determined according to the software testing type or software type.

The list of ALM and Test Automation Tools that are used and not preferred to be used in AKGÜN SOFTWARE are provided in Appendices A.

Guidelines to Select Specific Tool in Given Scenarios:

The guideline details that are recommended to be followed by companies in the process of supporting their systems with tools are provided below;

Our aim to prepare the Standard Tool Selection Guideline is to provide a guidelines for companies in their decision-making stage for supporting their systems with tools and in the steps of determining the selection criteria for the tools planned within this scope, and evaluating them.

This guideline is used, when the existing tools within the body of AKGÜN Software do not meet the needs or a tool searching process starts for a new need. In the meantime, a process should be followed in order to select the right tool.

ALM Tool Selection Stages:

Within the body of AKGÜN Software, the following steps are applied when investigating a new tool;

- **Problem Definition**
  It should be determined why this tool is needed in this step.
- **Determining User Needs**
  The keywords of the relevant field research should be determined by determining the user needs. In this step, the frame of the field research is narrowed.
- **Listing the differencese between the existing tool and investigated tools**
  It should be determined which needs are not met according to the user by the existing tool. It should be compared with the user need list. If the tool in question is a new tool to be incorporated into the body of AKGÜN SOFTWARE, the following step is followed.
- **Starting the field research process for the tools suitable for the needs**
  Market research is made in relation to the tool. The criteria are taken into consideration, such as determining price, version, and technical specifications, ensuring its integration with the tools used within AKGÜN Software, determining the level of meeting user need, determining price / performance etc.

- **Taking reference of tool selection criteria and multiplying the selection criteria, if needed**

  The determined criteria should be taken as basis in the tool selection process. If there is a criterion that is not in the criteria list and containing the related tool information, it should be added.
- **Evaluating with the scoring method with a reference to the tool selection criteria for each tool**

  The scoring method should be applied by taking the possibility of the tool's meeting the relevant criteria into consideration.
- **Starting meetings to get the most suitable tools for the criteria and make a demo (if any), otherwise starting use trial versions**

  Before making sure to purchase the tool, the company representative should be contacted and requested to make a demo. If a demo opportunity is not available, the free trial version process should be requested and started. Thus, the active implementation of the tool will enable the identification of an integration problem with the tools used within the AKGÜN Software and will provide a clear information about the tool and the right decision will be made.
- **Evaluating the suitable tools after demo**

  A list of the tools best suited to the scope of the organization should be determined by taking the criteria into consideration and by contacting and organizing a meeting with the relevant people. This list should be recorded with the minutes of the meeting.
- **Deciding the tool**

  The most appropriate decision should be made for the tool to be included in the AKGÜN Software and should be recorded with the minutes of the meeting.

ALM Tool Selection Criteria:

- Does the tool meet the use needs?
- Is the tool appropriate for the project environment and technology that is being used?
- Does it support all tools and objects that are used in the development stage in terms of integration?
- Does the tool provide free trial version to evaluate it before making a decision?

- Does the tool offer all existing specifications in the trial version?
- If the tool does not have a trial version, do company representative make a demo?
- Does it cover all specifications of the tool that is being actively used?
- How will the adaptation process of the tool be?
- Within the installation, is it possible to make a duration estimation?
- In terms of the progress of the project, is this transition period within an acceptable range?
- Is the tool project-based or is it being investigated for all projects within the scope of AKGÜN SOFTWARE?
  - If the tool is for a test automation, which test types does it support?
    - Unit test
    - Functional test
    - Regression Test
  - How user-friendly is to provide input test for complex or load tests? (Excel, XML, text document etc.)
- Does it provide strong reporting?
- Does it become integrated with other tools, such as project planning and test management tools?
- Is it possible to return the tool in case of unacceptable conditions?
- What are the comments/references of the existing customers for the tool?
- Does the seller give the first training?
- Is the support process flexible? Is only the first training support provided?
- If the tools is for requirements management, does it support more than one requirement format?
- When there is a change in requirements, is it powerful enough to manage this change in the best way?
- Is there an instruction and installation manual?
- Is maintenance support provided?

With **ALM solution,** the development stages of a software project become;

- Collaborative

- Transparent
- Productive

Thus, it increases the software quality in organizations and help teams work more effectively.

The Benefits that ALM Solutions Bring are:

- Make teams integrated only by one Platform,
- Eases the information flow,
- Make production process controllable,
- Time management can be done and Project planning become easier.

Among the ALM products, IBM JAZZ, on the other hand, comes with certain templates supporting software development processes. Agile & Scrum Processes, which are among the most recent software methodologies, are involved in these templates and cover agile software processes.

As explained within the scope of this study, HP UFT is used in order to test GUI (Graphical User Interface), API (Application Programming Interface), and multi-layered applications. HP Unified Functional Testing provides a software testing solution automized for problems that are due to the continual change in technology and processes. Automation is an important step in modern testing applications. It increases the software quality significantly by reducing the testing costs and complexity even in environments changing too much. With its integration with HP ALM, it ensures the developer and tester productivity and collaboration.

- Test automation procedures in an organization reduce the human-based application in manual tests, resulting in better quality of system testing.
- It increases productivity and efficiency in the software testing process by identifying more errors earlier.
- Automating repetitive tests reduces test cost.
- It plays an important role in the rapid completion of regression testing in infrastructure modification.
- It increases the percentage of code covered by tests (Code Coverage).
- It increases the quality of the developed product.

When we consider “why a test automation tool”;

Table 41: Manual and Automated Testing Advantages Table

Manual	Automated
Real Experience	Accurate
Lower Starting Cost	Fast
Detailed	Effective, Productive
	Continuous

In automation systems, the most important errors are found with Ad Hoc tests, i.e. the tester puts itself in the customer's place and tries various scenarios. Thus, it detects important errors and this is a possible feature in the HP UFT test automation tool.

The Problems and Obstacles We Have Encountered While Implementing ALM Tools:

Whereas it is an advantage that the tools selected during the commissioning process of ALM tools are compatible with our CMMI processes and agile methods, several problems were encountered, such as adaptation problems in the matching of requirements, problems of adaptation of workflow diagrams to the tool and, as a solution, transfer of flows to the tool by preparing them in Microsoft Visio, and transferring screen designs to the tool by drawing them in Visual Studio.

The ease of installation of the selected test automation tool (HP UFT) and its compatibility with our CMMI Test Processes are evaluated among the advantages. However, we have encountered a loss of effort and time due to the problem of its integration with IBM JAZZ and consequently in the process of searching for a third party software as a solution. On the other hand, the authorization limits of IBM JAZZ have also emerged as a separate obstacle.

CHAPTER 5

5.CONCLUSION AND FUTURE WORK

As a result of the research and examinations carried out by Gartner [46], the tools/suppliers used in software engineering and integrated solutions to meet ALM requirements are shown in the graph below, and the solutions chosen by AKGÜN SOFTWARE have resulted in the work with the market leaders in this regard and the best and quality tools that at the same time meet the needs at the maximum level have been preferred.

Figure 9: Magic Quadrant for Integrated Software Quality Suites [46]

Our action research study has been implemented to include a variety of departments within AKGÜN SOFTWARE. The results of the vast number of pilot applications have been analysed and the results have been compared with the the literature. We belive that the work we have done has a high validity. Choosing a single platform in a number of alternatives was a challenging process. It is not claimed that

the choosen ALM tool is the best out of all, however, it is the one that fits the defined criteria for our company. These criterias may be different for other companies depending on their needs. Similariy, a company with matching needs and features could select the same tool as a systematic and general approach have been used.

In this thesis, an action research study for ALM adoption in industrial context has been conducted. Action research studies are rarely conducted and this study is one of the very first ones that have been published, which is integrated and highly detailed. Our action research study offers a comprehensive experimental analysis and provides in-depth insight into ALM implementation. Action research results represent the studies conducted between the years from 2015 to 2018 in AKGÜN SOFTWARE. The outcomes of this action research is a very important guide for the companies with similar needs as it has very valuable lessons learned and experiences. Apart from adoption of a highly valuable and beneficial tool, we have also gained an in dept information about the various ALM versions and applications.

Eventhough this was our first action research, we were eager to provide a sustainable solution as we were aware of its importance. Our second and third action research cycles were based on the experiences and lessons learned from our first action research cycle. Hence, ALM 2.0 and 3.0 have been implemented in accordance with the outcomes of these action researches. The process and tool integration problems have been addressed with the implementation of ALM 2.0 and ALM 3.0. In a short amount of time, these ALM platforms have been cascaded whitin the company and have been implemented in other projects. With this study, ALM integration solutions for companies have been created.

This study has provided invaluable experiences and lessons not only for the problems of our company but the other companies as well. ALM tools have been evaluated by a numerous number of identified criteria. Also, selected instruments have been evaluated by the approach we have developped. The selected ALM tool have been smoothly implemented and integrated through pilot studies by using the systematic approach we have identified.

This study is a great milestone and a very important achivement for the company as the result of the study has provided a very valuable long-term benefits such as tracebility, integration and process automation. Therefore, development time and cost have been reduced, quality management processes have been improved and all the other processes have been managed in a harmony within the company. Although engineers, managers and some other stakeholders have resisted to the adoption of the ALM in the early stages, in the end they have realised the benefits of it. The ongoing problems of ALM 2.0 have been identified with the aid of critical research approach of the accepted action research and ALM 3.0 has been initiated. One of the major problems of ALM 1.0 and 2.0 has been identified as the lack of multi-platform solutions.

In software engineering adoption of action research for ALM is lately drawing more attention and becoming one of the most reliable research methodologies as opposed to the past. However, some studies suggest that action research studies are not mature enough as yet and they need to be improved. Our study can be used as a guide for the implementation of action research phases. Our firm has made some radical chages to implement this in a very short amount of time to add another success story to its achivements list. Later, our firm has been invited by IBM Jazz Business Partner to a conferance to share our success story with the other participants. Transition of a firm to IBM Jazz and comissioning within this scope wold normally require one year minimum but this process has taken only six months for AKGÜN SOFTWARE.

THREATS TO VALIDITY:

In this study, any events are observed by us as impartially as possible, and the applications and results achieved by us reflect the current situation. The factors jeopardizing validity of this study are as follows:

Assuming that the assessment results change according to the current environment, it cannot be alleged that the results of any research may not be generalized directly for another situation. However, it is clear that the results of this research may be implemented by any other companies in a similar organization and where there are any similar requirements, and it may be said that there may be the same or similar resolutions/results and thus this research study has an external validity under any

similar conditions and as long as their budgets are feasible. Most important factors and threats that will affect the external validity of this study include availability of the relevant competent resources in the organization, where the study will be conducted, support of the MANAGEMENT, feasibility of the budget, determination of making a relevant investment, and the processes developed according to the international standards such as CMMI, etc.

In the study, selection of a sample in a sufficient size is one of the factors affecting the internal validity, and we can say that, in this study, the pilot project selected in scope of ALM 1.0, ALM 2.0 and ALM 3.0 is a project of the company, where it has the widest customer portfolio and a number of qualified and large teams and also which uses the latest and most common technologies in the Company, and it has an internal validity. In the other hand, another factor affecting the internal validity is also that any subject is discussed by several researchers and more resources and opinions are applied. In this scope, more stakeholders of the Company, who are professional in their field, are included in this research, and any obtained findings are confirmed by means of consultations with the teams.

As Yilmaz states that [50] to deal with construct validity issues, we conducted a number of literature reviews also in this study. And secondly, we asked the another organizations/firms about their processes and tools.

The researches and studies show that ALM transformation and relevant ALM requirements give any expected results to ensure that the processes of a large scale company may run in an integrated and complete coordination way. However, any works and procedures such as determination of the requirements, matching of the requirements, design of the tests, establishment of the test scenarios and steps, traceability of the results on manual and automatic execution of the tests, correlating the test steps and test results with the requirements, and efficiency in the project management may be implemented fast and efficiently by means of ALM transformation. In the other hand, any results arisen from ALM transformation include any improvements such as facility and efficiency increase in quality of the product and application of the agile processes, an increase in the release speed of the product, shortening in the distribution processes of version and quality increase. Eventually, we can orchestrate a large-size software development organization by using ALM

transformation process and agile best practices. Eventually, we can orchestrate a large-size software development organization by using ALM transformation process and agile best practices.

As a result, we are confident to say that action research adoption is a highly useful methodology for the company in order to support the dissemination and the integration of ALM.

FUTURE WORK:

A new action researh in this context will be started in the company and the integration points within the scope of ALM 3.0 will be expanded. The action research studies will be continued in order for the requirements management, test management, error management and test automations to work in an integrated and a complete harmony with business and project management tools and processes. Action research studies will also increase traceability among these activities and allow us to manage the project management processes and software development lifecycle processes in a more integrated and effective manner.

REFERENCES

[1] İNÇKİ, Koray; SARIKAYA, Barış; ADİL, Türkay. Yazılım Projelerinde Uygulama Yaşam Döngü Yönetimi.

[2] LACHEİNER, Hermann; RAMLER, Rudolf. Application Lifecycle Management as Infrastructure for Software Process Improvement and Evolution: Experience and Insights from Industry. In: Software Engineering and Advanced Applications (SEAA), 2011 37th EUROMICRO Conference on. IEEE, 2011. p. 286-293

[3] MCHUGH, Martin; MCCAFFERY, Fergal; CASEY, Valentine; PİKKARAİNEN, Minna. Integrating Agile Practices with a Medical Device SDLC. In: European Systems and Software Process Improvement and Innovation Conference on EuroSPI, 2012. p. 1-6

[4] STOUT, Glenn A. Requirements traceability and the effect on the system development lifecycle (SDLC). Research Paper, Spring Cluster, 2001. p. 3-17

[5] BEYDAĞLI, Erkut; KARA, Mehmet; BAHŞİ, Hayretdin; ALPARSLAN, Erdem. Güvenli Yazılım Geliştirme Modelleri ve Ortak Kriterler Standardı. Ulusal Yazılım Mühendisliği Sempozyumu, 2009. p. 11-17

[6] KLESPITZ, József; BÍRÓ, Miklós; KOVÁCS, Levente. Aspects of improvement of software development lifecycle management. In: Computational Intelligence and Informatics (CINTI), 2015 16th IEEE International Symposium on. IEEE, 2015. p. 323-327.

[7] BŐRŐ, Miklós. Functional Safety, Traceability and Open Services. In: Software Engineering from Research and Practice Perspective. Scientific Papers of the Polish Information Processing Society Scientific Council. 2014. p. 73-82

[8] MICROSOFT. Driving Your Business Forward with Application Life-cycle Management. In: The State of Application Development in Enterprises and SMBs on Forrester Research. 2007. p. 1-8

[9] TÜZÜN, Eray; YILMAZ, Murat; ÜSFEKEŞ, Çağdaş. Uygulama Yaşam Döngüsü Yönetim Araçlarının Verimini Arttırmak İçin Ciddi Oyun Uygulaması. 2017. p. 158-168

[10] HERDEN, Sebastian; ZWANZIGER, André; ROBINSON, Philip. Declarative application deployment and change management. In: Network and Service Management (CNSM), 2010 International Conference on. IEEE, 2010. p. 126-133.

[11] PÍCHA, Petr; BRADA, Premek. Alm tool data usage in software process metamodeling. In: Software Engineering and Advanced Applications (SEAA), 2016 42th Euromicro Conference on. IEEE, 2016. p. 1-8.

[12] MAKSIMENKOVA, Olga; NEZNANOV, Alexey. Blended learning in software engineering education: The application lifecycle management experience with computer-supported collaborative learning. In: Interactive Collaborative Learning (ICL), 2015 International Conference on. IEEE, 2015. p. 655-662.

[13] KÄÄRIÄINEN, Jukka. Towards an application lifecycle management framework. VTT, 2011. p. 1-93

[14] An introduction to the benefits of Application Lifecycle Management. In: IKANALM. p. 3-11.

[15] GOTEL, Orlena, et al. The grand challenge of traceability (v1. 0). In: Software and Systems Traceability. Springer, London, 2012. p. 343-409.

[16] M.GROSS, John; MCINNIS, Kenneth R. Waterfall made simple: demystifying and applying Toyota's legendary manufacturing process. In: KANBAN, Made Simple. AMACOM Div American Mgmt Assn, 2003. p. 1-257.

[17] LİAO, Yongxin; LEZOCHE, Mario; LOURES, Eduardo Rocha; PANETTO, Herv´e; BOUDJLİDA, Nacer. A semantic annotation framework to assist the knowledge interoperability along a product life cycle. In: Advanced Materials Research, Trans Tech Publications. 2014. p. 424-429.

[18] ÖTLEŞ, Prof. Dr. Semih ; ATALAY, Prof. Dr. Süheyda; GÜNEŞ, Prof. Dr. Semih; ERTEKİN,Prof. Dr. Figen; YILDIZ, Prof. Dr. Hasan; SAYER, Doç. Dr. – Ing. Sami; ÖZDEN, Doç. Dr. – Ing. Hüseyin; ALVER, Doç. Dr. Ninel; YEŞİLAY, Yrd. Doç. Dr. Rüstem Barış; BULUT, Yrd. Doç. Dr.

Hasan. Ürün Yaşam Döngüsü Yönetimi (PLM). Ürün Geliştirme, Ege Üniversitesi PLM Proje Grubu. Ocak- Şubat 2015. p. 36-45.

[19] TAYA, Sanjana; GUPTA, Shaveta. Comparative Analysis of Software Development Life Cycle Models. In: Dept. of computer applications, Seth Jai Parkash Mukand Lal Institute of Engineering & Technology. IJCST Vol. 2, Issue 4. Oct. - Dec. 2011. p. 536-539.

[20] JUNLING, Liang. Developing CRM System of Web Application based of JavaServer Faces. In: Second International Workshop on Education Technology and Computer Science. 2010. p. 766-769.

[21] SİMAO, Elisio Maciel. Comparison of Software Development Methodologies based on the SWEBOK. October 27, 2011. p. 1-95.

[22] SCHWABER, Carey. The Changing Face of Application Life–Cycle Management. In: Tomorrow's ALM Platforms Will Deliver On The Promise Of Today's ALM Suites. 2006. p. 1-13.

[23] ROSSBERG, Joachim. Introduction to Application Lifecycle Management. In: Beginning Application Lifecycle Management. Apress, Berkeley, CA, 2014. p. 13-32.

[24] LOGICAL, Commercial. Medical devices—Application of risk management to medical devices. 2007.

[25] Microsoft. Application Lifecycle Management with Visual Studio and Team Foundation Server, Retrieved on: 28 January 2017, Available on: https://msdn.microsoft.com/en-us/library/fda2bad5(v=vs.110).aspx

[26] INTERNATIONAL ORGANIZATION FOR STANDARDIZATION; INTERNATIONAL ELECTROTECHNICAL COMMISSION. Software Engineering--Product Quality: Quality model. ISO/IEC, 2001.

[27] FRAKES, William; TERRY, Carol. Software reuse: metrics and models. ACM Computing Surveys (CSUR). p. 415-435. 1996.

[28] CHAPPELL, David. et al. What is application lifecycle management Article Chappell & Associates, 2008 on http://davidchappell.com/writing/white_papers/What_is_ALM_v2.0--Chappell.pdf

[29] CHAPPELL, David. et al. Application Life-Cycle and Business Strategy Article on 2010 on

http://www.davidchappell.com/writing/white_papers/ALM_and_Business_Strategy_v2.0--Chappell.pdf.

[30] CHAPPELL, David. et al. Application Life-Cycle and Business Process Article on 2010 on http://www.davidchappell.com/writing/white_papers/ALM_as_a_Business_Process_v2.0--Chappell.pdf

[31] CHAPPELL, David. et al. Business Value of Agile Development Article on March,2012 on http://www.davidchappell.com/writing/white_papers/Business_Value_of_Agile_Development_1.0--Chappell.pdf

[32] WILSON, W. et al. Magic quadrant for application development life cycle management. Gartner, February, 2015.

[33] GARTNER. Application Life Cycle Management. Retrieved on: 20 December 2018, Available on: https://www.gartner.com/reviews/market/application-development-life-cycle-management

[34] BASKERVILLE, Richard L. Investigating information systems with action research. Communications of the AIS, 1999, 2.3es: 4.

[35] KRAVCHİK, Moshe. Application Lifecycle Management Environments: Past, Present and Future, Thesis of M.Sc. degree in Computer Science The Open University of Israel, 2009.

[36] ERYILMAZ, Ömer. TFS-1: ALM (Application Lifecycle Management), Retrieved on: 28 January 2017, Available on: http://www.omereryilmaz.com/alm-application-lifecycle-management-nedir/

[37] CORNE. Caliber 10.1 is here!. Retrieved on: 28 January 2017, Available on: https://community.microfocus.com/borland/define/caliber_-_requirements_management/b/weblog/archive/2012/03/01/caliber-10-1-is-here.aspx

[38] BORLAND. Retrieved on: 28 January 2017, Available on: https://en.wikipedia.org/wiki/Borland

[39] STAR TEAM. Retrieved on: 28 January 2017, Available on: https://en.wikipedia.org/wiki/StarTeam

[40] MARCEL. Silk Central. Retrieved on: 28 January 2017, Available on: https://community.microfocus.com/borland/test/silk_central/b/weblog/archive/2014/07/02/released-silk-central-15-5.aspx

[41] KRİLL, Paul. Borland prepares to run Gauntlet for ALM, Retrieved on: 28 January 2017, Available on: http://www.infoworld.com/article/2658290/application-development/borland-prepares-to-run-gauntlet-for-alm.html

[42] IBM, IBM Rational DOORS vs IBM Rational RequisitePro. Retrieved on: 28 January 2017, Available on: https://www.itcentralstation.com/products/comparisons/ibm-rational-doors_vs_ibm-rational-requisitepro

[43] Open Services for Lifecycle Collaboration. IBM Rational Focal Point, Retrieved on: 28 January 2017, Available on: http://open-services.net/software/ibm-rational-focal-point/

[44] IBM. Rational ClearQuest and Rational ClearQuest ALM Appliance, Retrieved on: 28 January 2017, Available on: http://www.ibm.com/developerworks/downloads/r/rcq/

[45] IBM. Rational Rose Modeler, Retrieved on: 28 January 2017, Available on: http://www-03.ibm.com/software/products/en/rosemod

[46] WILSON, W., et al. Magic quadrant for application development life cycle management. Gartner, February, 2015, 9.

[47] CRESCENTINI, Alberto; MAINARDI, Giuditta. Qualitative research articles: guidelines, suggestions and needs. Journal of workplace learning, 2009, 21.5: p. 431-439.

[48] MUNĐAR, Dušan; MATOTEK, Darko; JAKUŠ, Marija. Quantitative research methods participation in the information sciences papers in Croatia. In: 23rd Central European Conference on Information and Intelligent Systems, CECIIS. 2012.

[49] KAJFEZ, Rachel Louis. The Motivation and Identity Development of Graduate Teaching Assistants in First-Year Engineering Programs. 2013. PhD Thesis. Virginia Tech.

[50] YILMAZ, M. "A Software Process Engineering Approach to Understanding Software Productivity and Team Personality Characteristics: An Empirical Investigation", PhD Thesis, Dublin City University (January 2013)

[51] ROYCE, W; “Managing the Development of Large Software Systems”, 1970. p. 328-338.

[52] KÄÄRIÄINEN, Jukka; VÄLIMÄKI, Antti. Applying application lifecycle management for the development of complex systems: Experiences from the automation industry. In: European Conference on Software Process Improvement. Springer, Berlin, Heidelberg, 2009. p. 149-160.

[53] AYTEKİN,Ahsen İkbal; TÜZÜN, Eray; Macit, YAGUP; TEKİRERDOĞAN, Bedir; “Uygulama Yaşam Döngüsü Yönetimi - Sistematik Eşleme Çalışması. In: Proceedings of the 9th Turkish National Software Engineering Symposium (UYMS 2015) p. 134–145.

[54] Qast Software Group. Products for the Software Development Life Cycle (SDLC), Retrieved on: 28 January 2017, Available on: http://www.qast.com/eng/product/develop/borland/index.htm

[55] S. Easterbrook, J. Singer, M.-A. Storey, and D. Damian, “Selecting Empirical Methods for Software Engineering Research,” Guide to Advanced Empirical Software Engineering, pp. 285–311, 2008.

[56] Test Management Tools, Retrieved on: 03 February 2019, Available on: https://www.itcentralstation.com/categories/test-management-tools.

PART I

APPENDICES A

Appendices

Test Automation Tools and ALM Platforms Usage Statuses in AKGÜN SOFTWARE

Group	Tool Name	Using Status in AKGÜN SOFTWARE (Yes/No)
Test Automation Tools	Selenium Web Driver	No
	Hp QTP – UFT	Yes
	Hp LoadRunner	No
	Sencha Test Studio	No
	Siesta	No
	Microsoft Test Manager	No
	Oracle Test Suite	No
	Silk Performer	No
	Sahi	No
	Selenium IDE	No
	Apache Jmeter	Yes
	SoapUI	No
	IBM Rational Test	Yes
ALM Platforms	Bamboo	No
	Bitbucket	No
	Clover	No
	Crucible	No
	FishEye	No
	JIRA Software	Yes
	IBM Cloud - PaaS (Bluemix)	No
	IBM Rational ALM	Yes
	Jama	No
	AgileCentral	No
	Polarion ALM	No
	Perforce Software Helix	No
	PTC Integrity	No

Group	Tool Name	Using Status in AKGÜN SOFTWARE (Yes/No)
	Kovair ALM Studio	No
	SUSE Studio	No
	SpiraTeam	No
	SwiftALM	No
	Microsoft Developer Network	No
	Team Foundation Server	No
	Rocket Aldon	No
	Rocket Software	No
	CollabNet CloudForge	No
	CollabNet TeamForge	No
	AccuRev	No
	Caliber	No
	Micro Focus (HPE Software) Application Lifecycle Management	No
	Micro Focus (HPE Software) Application Lifecycle Management Octane	No
	Silk Central	No
	StarTeam	No