Systematic Agile-Waterfall Hybrid Model to Manage ITSM Implementation Projects

Abdullatif A. Al-Zahrani[1]
Upstream Computer Operations Department
Upstream Computer Center
Dhahran, Saudi Arabia

Jassim A. Al-Selham[2]
Upstream Computer Operations Department
Upstream Computer Center
Dhahran, Saudi Arabia

Safran A. Al-Safran[3]
Upstream Computer Operations Department
Upstream Computer Center
Dhahran, Saudi Arabia

Abstract:- The vast advancements of modern Information Technology Service Management (ITSM) solutions, which cover a wide spectrum of business processes, have resulted in increased customer expectations, which are translated into complex functional/ non-functional requirements for ITSM implementation projects. Typically, implementing such complex requirements involves an experienced third-party vendor, who eventually imposes logistical constraints that limit the use of the entire Agile model and steers the project towards the traditional Waterfall model. On the other hand, there are legitimate reasons for employing the Agile model. These reasons are either related to better management of requirements analysis, handling challenges in the new ITSM solutions especially with continuously added features, or addressing unanticipated events during the project execution.

This paper presents a systematic approach to employ the Agile-Waterfall hybrid model for better efficiency throughout the ITSM implementation project phases. The model classifies the requirements into two types: (1) customer functional/non-functional requirements and (2) ITSM solution-specific requirements. Based on their initial analysis, the model creates structured information about each requirement which facilitates its implementation or future analysis, if needed. Additionally, it utilizes the Agile Change Management process, with pre-defined criteria, to address requirement analysis activities that result from unexpected events during the implementation phase. With a balance between flexibility and control, the end result is a verified implementation meeting all project requirements, utilizing advanced ITSM features, with solid documentation, and satisfied parties.

Keywords:- *ITSM, Agile-Waterfall Hybrid, Requirement Analysis, Information Technology Service Management, Project Management.*

I. INTRODUCTION

ITSM solutions were developed to assist IT organizations in managing and delivering their services more efficiently, with improved quality, better controls, and lower costs. To achieve this, ITSM solutions provide a wide range of integrated capabilities such as discovering and managing IT assets, managing customers' requests and issues, controlling IT changes, monitoring IT services availability, capturing and managing IT knowledge, tracking compliance with customer Service Level Agreements (SLA), automating self-services, customer surveys, performance reports, and others. Modern ITSM solutions, which utilize latest software deployment techniques, employ Artificial Intelligence (AI) to further gain more added values.

In order to meet their purpose, both implementation and day-to-day administration of ITSM solutions should be easy and manageable with minimum resources and efforts. Otherwise, their purpose is defeated and instead they become overhead. Implementing an ITSM solution requires running a project with defined scope/deliverables, schedule, and resources. ITSM implementation projects, like any other projects, can use either the traditional Waterfall model or the Agile model. However, "It's really rare that a project would fit perfectly into a pure agile or waterfall approach. Most, rather, fall on a spectrum between the two". [5]

II. NATURE OF ITSM IMPLEMENTATION PROJECTS

Understanding the nature of the ITSM implementation project helps in choosing the best implementation model. ITSM implementation projects share lots of similarities with other IT projects, however, they have the following specific characteristics and facts that make them different:

➢ *Large Customer Requirements and Expectations*

ITSM solutions have a diversity of modules and capabilities covering many IT business processes and attracting users and management personnel. IT users, who interact on a daily basis with ITSM solution whether directly or indirectly, have very high expectation especially with the trending automation, self-services, dashboards, chatbots, mobility, AI adoption and others. On the other hand, IT managers have very high expectations especially when it comes to reduction of operational costs and IT staff, increasing customer satisfaction, and coping with modern IT technologies. As requirements increase, their analysis efforts increase with some challenges that may lead to scope creep, exceeding budget, poor design and functionality, and rework. [1]

➢ *3rd Party Implementation Vendor Involvement in ITSM Projects*

Since ITSM solutions require specialized skills and experience, most of the time IT organizations elect to engage a 3rd party vendor for implementation according to defined requirements within a scope of work. Engaging an implementation vendor is also a fact seen by senior IT professionals who understand the level of required skills and specialties. Vendor contract's fixed terms and conditions, which are defined prior to project implementation, control the compliance with the project scope of work. Thus, as a common practice to avoid deviating from agreed upon contracts, all known requirements, derived from the scope of work document, are analyzed prior to the project's start, and implementation plans are developed and considered final. With this approach, chances of having an optimal implementation of the ITSM solution, with its advanced features, are low. Eventually, the impact on the project is less flexibility on both project schedule and modifications to the scope of work.

➢ *Engagement of users in the ITSM Project is less than Required*

Although ITSM solutions are intended to manage services delivered to them, most IT users perceive ITSM systems as IT-internal systems and may not be very excited to be heavily engaged during the project especially in the implementation phase or continuous requirement analysis. The less involvement of users impacts requirement analysis and testing, which affects quality of project deliverables.

➢ *Enhancements are Continuously Added in Modern ITSM Solutions*

Enhancements in ITSM solutions are continuous especially with the advanced rapid software development technologies and enablers. This means software limitation or unavailable features today are potentially available in the near future, even during the project timeframe. Having continuous enhancements in modern ITSM solutions increases the chances to modify project plan to adopt new features.

➢ *Medium to High Level Customizations Always exist in ITSM Projects*

ITSM requirements analysis may end up with a moderate to high level of customizations so that they meet business needs if there is no available out of box capability. "Even though an out-of-the box solution provides a starting point, successful ITSM solution implementation is achieved by customizing the solution to match the applicable enterprise practices. [4] Implementation of customizations may add more complexity to project and adds more unknowns.

➢ *ITSM Solutions have many integrations with External Systems*

ITSM solutions, by nature, have many integrations with other external IT systems to enable them meet their enterprise-wide purpose. Examples include integration with HR systems, authentication systems, asset repositories, service portals, and others. This imposes a mandatory dependency which has its own consequences.

➢ *ITSM Solutions Impose their own Way*

ITSM solutions come with their own way of performing certain tasks and processes even if they are compliant with ITIL standards. This definitely imposes additional ITSM solution-specific requirements which logically change the original customer requirements. A good example of an ITSM solution-specific requirement is a new feature that cannot be eliminated and, at the same time, is not needed by the customer. Definitely, such situations lead to assessments and discussions with customers to make the proper decision especially if there are multiple dependencies on this feature. ITSM solution-specific requirements should be discussed/analyzed and considered as part of the project plan.

➢ *Quick Deliverables at Early Stages of the Project Timeframe*

Completion of implementation of some relatively straightforward ITSM modules could be achieved at an early stage of the implementation phase. This provides a means to perform partial UAT during the project timeframe. Performing UAT tests increases the chance of finding issues that end up in a re-analysis of requirements.

The above facts are summarized in the below figure that shows their impact on the steering the project to either the Agile or the Waterfall model.
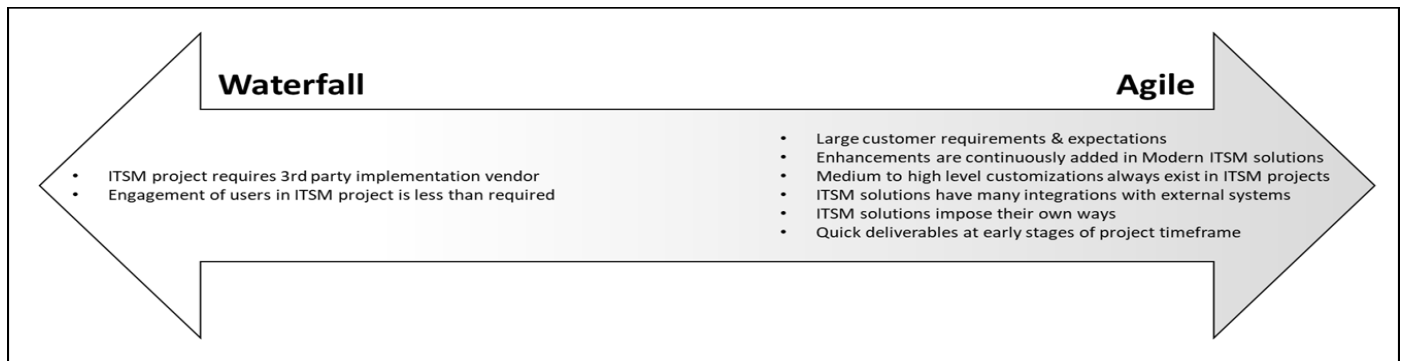
Fig 1 Impact of Nature of ITSM Projects on Project Model

From the previous facts of ITSM implementation projects, it is obvious that some requirements may take a long time to analyze, change of plans (for any reason) may happen during the project implementation stage, and more controls need to exist in the project. With its combined benefits, the Agile-Waterfall Hybrid model can manage all these implications and drive the project towards success. Issues such as budget incompliance, poor design and functionality, rework during implementation, and scope creep are minimized.

III. OVERVIEW OF AGILE, WATERFALL, AGILE-WATERFALL HYBRID MODELS

Both Waterfall and Agile models are commonly used in IT projects. "A waterfall model project has a fixed schedule and budget and involves a sequence of defined steps to build a project. An agile model project has an evolving scope and is designed to be flexible and responsive to changing conditions. Both process models are based on the idea that each task should be broken down into smaller steps." [6] Table 1 shows comparisons between both Waterfall and Agile models.

Table 1 Comparison between Waterfall and Agile Models [2], [6]

Factor	Waterfall	Agile
Process orientation and flow	Linear/sequential	Iterative
Scope	Defined in advance	Flexible to scope changes
Plan	Plan-driven	Open to modify plans
Cost	Fixed and known in advance	Non-fixed funding and not easy to estimate. Works best with time and material funding
Schedule	Fixed	Flexible
Customer involvement	Mostly during User Acceptance Test phase	Continuous engagement of customers for feedback
Deliverables	Big bang	Continuous deliverables over iterations
Documentation	Extensive	Less documentation
Project progress tracking	Through the fixed project implementation plan	Not straightforward due to changes
Team & collaboration	As per the defined project roles	Highly needed
Suitability	Works best with well-known requirements upfront	Works well with dynamic environment with more flexibility of cost, customer engagement, and change-open culture

Due to their differences, some organizations use the Agile-Waterfall hybrid model to achieve an exceptional output in their projects. [6] The Agile-Waterfall hybrid model utilizes wisely the benefits of both Agile and Waterfall. According to Lucid Content Team [8], it is up to the organization to determine the best hybrid scenarios. Some of the scenarios include:

- Using Agile at enterprise level while using Waterfall for requirements, design and implementation
- Using Agile at project level, while using Waterfall at the organizational level
- Using Waterfall at both project and enterprise level, while using Agile for individual teams
- Using Waterfall at both team and enterprise level, while using Agile for specific project phase

Another way to use the Agile-Waterfall hybrid model is when a project has two different components such as a project with (equally important) hardware and software components or a project with software that has back-end and front-end technology. [7] The logic is that hardware or backend technology are usually implemented with standard (almost fixed) requirements and hence suitable for Waterfall. Software (or front-end technology) deployments, on the other hand, are implemented through Agile since they have a large number of customer-facing requirements.

With the above suggested methods to use the Agile-Waterfall hybrid model, it is clear that there is no specific way and it depends totally on the organization's needs. Hence, looking from a different angle, a similar analogy can be generalized on the analysis of project requirements.

There are straightforward requirements that can be analyzed and implemented directly through the Waterfall model and there are requirements that are best managed by the Agile model for good reasons. Considering their large number of requirements along with their specific nature, ITSM implementation projects (especially if it is a migration from an old ITSM solution) are good candidates for using the Agile-Waterfall hybrid model.

IV. SYSTEMATIC AGILE-WATERFALL HYBRID MODEL TO MANAGE ITSM IMPLEMENTATION PROJECT

In this section, the Agile-Waterfall hybrid model is explained. For better understanding, it is important to explain the standard phases of the ITSM implementation project first, since the hybrid model is a modified version of it.

A. Standard Phases of ITSM Implementation Projects

After choosing an ITSM solution and completing the required logistics to engage an implementation vendor, the ITSM implementation project starts. The standard phases of the ITSM project are the following:

➤ *Requirement Analysis & Design*

This phase is important to further detail the scope to a list of implementable tasks. It involves discussion and analysis of both types of requirements, prototyping, and architecting the final solution design to meet IT organization needs. Additionally, the project implementation plan is finalized in this phase.

➤ *Implementation & Verification*

This phase executes the tasks in the project implementation plan and ensures verification of implementation. Issues during this phase are expected as things may not work as planned and hence issues are managed accordingly.

➤ *User Acceptance Test*

This phase involves user testing in a semi-production test environment. Testing specific modules of ITSM solutions can be done separately, if applicable, or fully otherwise with the rest of modules. There are different types of tests: functional, performance, security, and operational (including BC/DR). There are other activities performed in this phase such as user training and knowledge transfer to the IT organization team.

➤ *Deployment & Go-live*

Once testing is completed and all issues are fixed, a final architected ITSM solution is deployed on the production environment. Then, the go-live can start either in a gradual or big-bang approach depending on the IT organization preference.

B. Employing Agile-Waterfall Hybrid Model

The Agile-Waterfall hybrid model, tailored mainly for handling requirements in ITSM implementation projects, is shown in Figure 2. In short, it utilizes the controls of the Waterfall model throughout the project and employs the Agile model for handling requirement analysis during the first three phases. The end result is that the implementation is continuing while other requirements are being analyzed and being prepared for implementation.
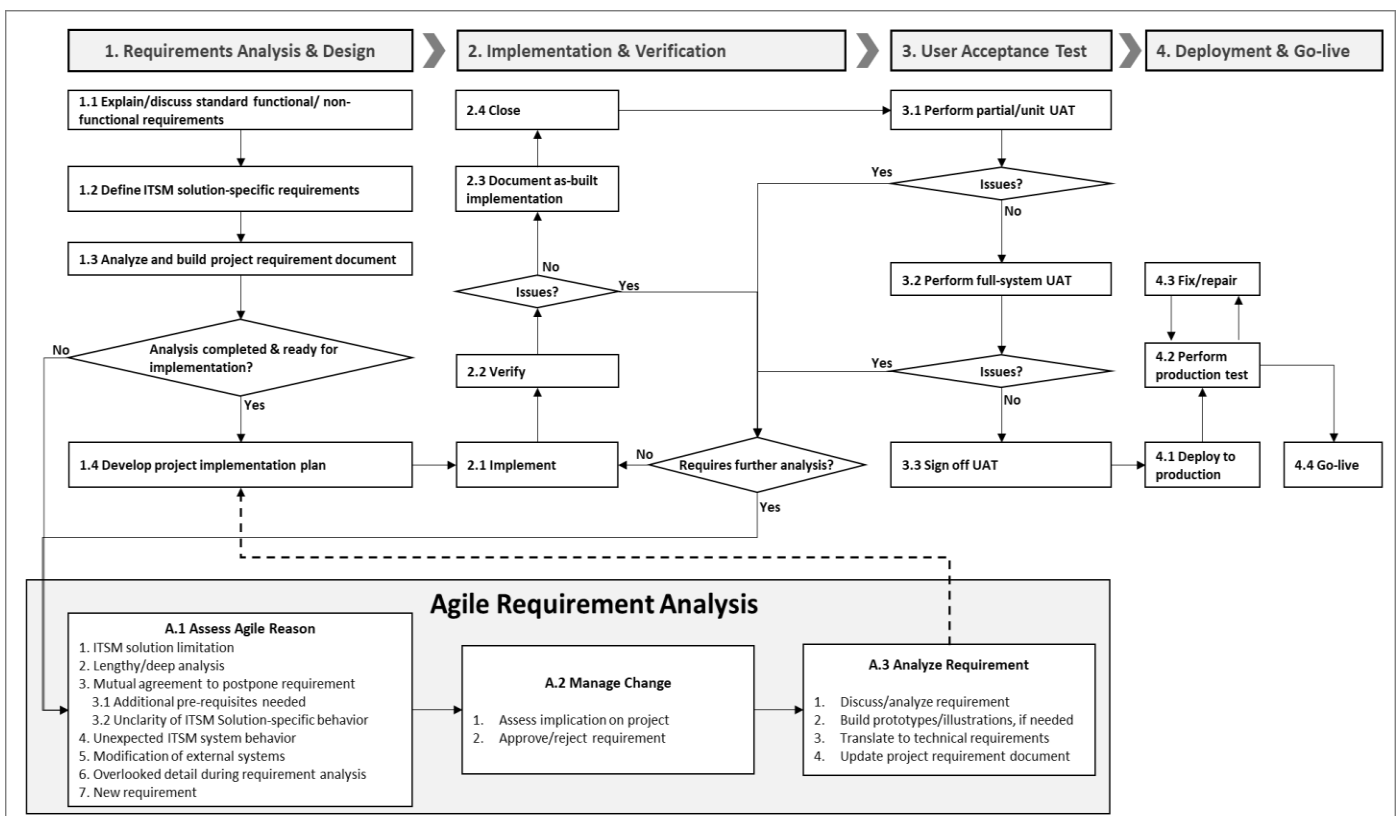

Fig 2 Systematic Agile-Waterfall Hybrid Model to Manage ITSM Requirements

> *The Table below Summarizes the Steps in the Agile-Waterfall Hybrid Model.*

Table 2 Steps in the Systematic Agile-Waterfall Hybrid Model

Phase	Process	Agile Process		
1. Requirement analysis& design	1.1 Explain/discuss customer functional/ non-functional requirements	A1. Assess Agile Reason	A.2 Manage Change	A.3 Analyze Requirement
	1.2 Define ITSM solution-specific requirements			
	1.3 Analyze and build project requirement document			
	1.4 Develop project implementation plan			
2. Implementation & verification	2.1 Implement			
	2.2 Verify			
	2.3 Document as-built implementation			
	2.4 Close			
3. User Acceptance Test	3.1 Perform partial/unit UAT			
	3.2 Perform full-system UAT			
	3.3 Sign off UAT			
4. Deployment & Go-live	4.1 Deploy to production			
	4.2 Perform production test			
	4.3 Fix/repair			
	4.4 Go-live			

C. Below are the Explanations of each Step in each Phase of the Model:

> *Phase 1: Requirement Analysis & Design*

Prior to the start of the ITSM implementation project, it is essential that the IT organization identifies the customer functional/ non-functional requirements. This phase determines the subsequent activities starting from selecting a suitable ITSM solution, developing the scope of work along with the required contract, defining project acceptance criteria, and allocating budget. The functional requirement simply refers to what the system should do; while the non-functional requirement refers to how the system should behave. Sending a notification upon closure of a request is an example of functional requirement while high availability of the ITSM portal is an example of the non-functional requirement. The customer functional/ non-functional requirements are usually developed based on understanding practices from the ITIL (Information Technology Infrastructure Library) framework or by surveying different implementations of ITSM solutions. It is, apparently, written in a business language not specific to any ITSM solution. Eventually, it will be the basis for the scope of work document on which the contract will be developed.

It is highly recommended that the IT organization forms an implementation team to work with the implementation vendor. This team is technically engaged in the project and coordinates any required logistics for the project.

- *Explain/Discuss Customer Functional/ Non-Functional Requirements:*
Once the contract logistics are completed and the project officially starts, the customer functional/ non-functional requirements are clarified and explained through discussion sessions with the implementation vendor during this step.

- *Define ITSM Solution-Specific Requirements:*
During the requirement analysis phase, the implementation vendor is expected to demonstrate and explain, to the IT organization team, the ITSM solution, its features, capabilities, behaviors, user interface…etc. Though ITSM solutions providers claim that their solutions are compliant with ITIL, their ITSM implementation and functionality differ from one solution to another. In fact, ITIL is not a standard, and it is a mistake to use the term "ITIL compliant". [3] A good example of ITIL differences is the implementation of change management where change types (classes), roles, approvals, and lifecycle vary among different ITSM solutions. There are also additional differences native to the ITSM solution itself, ranging from the use of relative timestamp (rather than absolute) to the complete look and feel of the user interface form.

At this step, these differences eventually trigger new requirements that are specific to the ITSM solution itself. These requirements might not be large compared to the customer functional/ non-functional requirements, however, they are very important to address during this step to avoid potential disagreements and rework during the implementation phase. The challenging part of these ITSM solution-specific requirements is that the IT organization team might not have sufficient experience to react and define a fixed requirement for a given behavior or capability at this stage. Hence, logically some of the ITSM solution-specific requirements are expected to be discussed in detail at a later phase and a proper decision is made accordingly.

- *Analyze and Build Project Requirement Document:*
Once all requirements are identified and documented in the last two steps, analysis starts by first categorizing, grouping, prioritizing, translating each requirement to technical terms, which may involve prototyping/modeling and illustrations, and finally validation by the IT organization/customer. One of the key deliverables of this step is to have a common understanding of the requirements between all involved parties. There are reasons that may

cause some challenges when analyzing requirements which eventually impose defining plans on how to handle them throughout the project. These reasons are the main drive for utilizing the Agile model. At the end of this step, all requirements should be written in a project requirement document (either in spreadsheet table or formatted document) containing at least the following information (as table columns):

✓ Reference # (Number) to identify each requirement
✓ Requirement description: text written in ITSM solution terminology for each requirement
✓ Expectation: text describing the expected behavior/outcome/configuration
✓ Requirement and expectation validated by IT organization/customer? (Yes/No)

✓ Priority (High, Medium, Low)
✓ Dependency (on other Reference #)
✓ Implementation efforts (High, Medium, Low) – see Table 3
✓ Analysis completed and ready for implementation? (Yes/No)
✓ Implementation approach (OOB configuration, Customization, 3rd party tool)
✓ Agile reason, if applicable. (One choice of the defined Agile reasons)

The field "implementation efforts" uses criteria that should be agreed on between the implementation vendor and IT organization team. As a starting point, the below criteria could be used (or modified to suit project needs):

Table 3 Proposed Criteria for Estimating Implementation Efforts

Items	High	Medium	Low
Customization	Huge New and not tested before	Moderate Known & tested customization	No
Duration	More than 7 days	Between 3 to 7 days	Less than 3 days
Re-work/re-design of dependent requirements	Needed, major	Needed, minimum	No

If there are new requirements (not defined in the project implementation plan) that surface throughout the project, implementation cost should be agreed upfront for each category as follows:

Table 4 Proposed Template for Cost of New Requirements based on its Implementation Efforts

Cost	High	Medium	Low
Cost of new requirement	$$	$	0

The field "Analysis Completed and ready for implementation?" is very important for the Agile model. In this step, if it is "No", then the Agile reason must be set to one of the following reasons suitable for this phase:

➢ *Reason 1: Limitation in ITSM Solution to Meet Requirement*
Some requirements (mostly from the customer functional/ non-functional requirement) impose utilizing certain features, functionalities, or capabilities that do not exist fully or partially in the ITSM solution (or its current version). Chances of having such features during the implementation phase are possible with the rapid enhancements to ITSM solutions.

➢ *Reason 2: More Time is Required to Perform Lengthy/Deep Analysis*
Some requirements require deep analysis, which involves prototyping and design. This may consume additional time and hence delay the start of the implementation phase. If it is not a core requirement, this situation suggests saving time and delaying its analysis during the implementation phase. An example of such requirements is the design and modeling of a totally new service that is based on a fulfillment of backend workflow with approvals and conditions.

➢ *Reason 3: Mutual Agreement to Delay Analysis on Specific Requirements*
In some situations, the IT organization team and ITSM implementation vendor realize the following reasons that delay analysis for certain requirements:

• There is a need for additional pre-requisite for prototyping and it is not available at requirement analysis phase. If the pre-requisite will be available during the implementation phase, it is wise to save time and delay such analysis.
• Minor or unclear ITSM solution-specific requirement that will be clearer during the implementation phase after gaining more insight into the ITSM solution. An example that could be seen during the ITSM solution technical sessions would be the change management pre-defined (or mapped) approval. The IT organization team may not be able to decide the approval configuration if they have insufficient understanding of how the approval is implemented in the ITSM solution. Hence this ITSM solution-specific requirement can be delayed until it is clear to make the right decision.

The next step is based on the information about each requirement in the project requirement document. Requirements that have been analyzed fully and are marked as implementable will proceed to the next step (1.4). Otherwise, they will proceed to the Agile Requirement Analysis along with the defined Agile reason.

➢ *Develop Project Implementation Plan:*
Requirements that reach this step are implementable. In this step, the information below is added to the project requirement document and updated accordingly for each implementable requirement:

- Requirement fulfillment tasks: (high level steps or description of what needs to be done)
- Required resources: names of the resources who perform implementation tasks

- Implementation duration

- *The Final shape of the Project Requirement Document Looks Like Table 5.*

Table 5 Project Requirement Document Template

Reference #	Requirement description	Expectation	Requirement & expectation validated by IT organization?	Priority	Dependency	Implementation efforts	Analysis completed & ready for implementation?	Implementation approach	Agile reason	Requirement fulfillment tasks	Required resources	Implementation duration

- *The Project Implementation Plan is Developed and Fed mostly by the Information in the Project Requirement Document.*

➢ *Phase 2: Implementation & Verification*

In this phase, the project implementation plan is executed, implementation is verified, and the as-built documentation is captured and updated. These steps are required to maintain the Waterfall controls.

- *Implement:*

This step is the actual implementation of the requirement which is performed by the ITSM implementation vendor and according to the project implementation plan. It is important to realize that since ITSM solutions have multiple modules, implementation of some of them may finish before others and hence partial testing can be done.

- *Verify:*

Verification of implementation is a quality control that is considered highly recommended. It should be done by the IT organization team as they are the beneficiary of the project. If verification succeeds, implementation of the requirement is marked as successful. If there are any issues, they are assessed and either go back to the previous step for fixing or go to Agile Requirement Analysis. At this point and regardless of how comprehensive the project implementation plan, there could be additional reasons to use Agile:

- *Reason 4: Unexpected ITSM System Behavior*

During the implementation phase, it is a fact that some of the implementation tasks may not work as planned initially due to system behaviors. This may lead to a complete review and analysis of the requirement to fix the situation.

- *Reason 5: Modification of External Systems*

In parallel with ITSM implementation projects, there are other external systems that have their own projects as well. It is likely to have an integration plan with an external system that might not be valid anymore during the implementation phase. Thus, re-analysis of the requirement is required so that a new proper design is built.

- *Reason 6: Overlooked Detail During Requirement Analysis*

In rare situations, there are (very specific) requirements that have not been addressed in the requirement analysis and design phase and suddenly surfaced during implementation. Re-analysis of such requirements is still required.

- *Reason 7: New Requirements (Replacement or Additional)*

During the implementation phase, new requirements may surface either to replace/modify/delete previously agreed upon requirement or to create totally new functionalities. The main reasons for these new requirements are:

✓ Uncommunicated features and capabilities of ITSM solution during the requirement analysis and design phase. Instead, they are discovered at this stage and some of them have great value. These features may change the original design or could be an eye-opener for other potential enhancement requirements.
✓ New requirements driven by business needs that were not ready during the requirement analysis and design phase. In this case, the requirements are subject to analysis and are evaluated according to the pre-defined implementation efforts and associated cost criteria (in step 1.3).

- *Document as-Built Implementation:*

To maintain Waterfall requirements, this step captures and documents the as-built implementation which is necessary for ITSM implementation projects and for future support.

- *Close:*

The implementation closure means that the requirement has been implemented as designed.

➢ *Phase 3: User Acceptance Test*

The User Acceptance Test (UAT) phase is where real tests by real users are performed. The steps below exist in this phase.

- *Perform Partial/Unit UAT:*

Prior to performing any UAT, user training should be provided to testers. Then, the implementation team can assess the best way to conduct partial/unit UAT prior to the full-system UAT. If a partial test is possible, it is very effective to capture issues on small ITSM modules before the readiness of the full ITSM solution implementation. An example of a partial (or unit) test is testing the incident management module alone without any interaction with any other module. If there are no issues, the test is marked as successful and ready for the full system tests. Otherwise, issues are assessed and will either go back to implementation or to Agile Requirement Analysis. Agile reasons in step 2.2 may surface in this step too.

- *Perform Full-System UAT:*

The full system test is conducted when that all requirements are implemented as designed and the ITSM solution is in operation in the test environment. In this step, the whole system is tested by users as if it is in the production environment. An example of a full-system test case is the creation of a user issue from a service portal which then is translated to an incident record and linked to the user asset. Once resolved within the defined SLA target, the user is notified and a knowledge article is created. If there are no issues, the test is marked as successful. Otherwise, issues are assessed and will either go back to implementation or to Agile Requirement Analysis. Agile reasons in step 2.2 may surface in this step too.

- *Sign off UAT:*

If all tests pass, a sign-off on UAT is important prior to going to deployment and go-live phase.

➢ *Phase 4: Deployment & Go-Live*

The deployment & go-live phase is the last phase of the project and hence no more development or requirement analysis activities exist. Accordingly, the Agile Requirement Analysis process is not active at this phase of the project as shown in Figure 2.

- *Deploy to production:*

In the best scenarios, deployment is simply mimicking the test environment to the production environment. However, if there are any differences between the two environments, the project implementation team should understand the differences and their implication and act accordingly. In an ideal situation, the differences are known and the required work is already in the project implementation plan.

- *Perform production test:*

Production tests are important to ensure readiness of the environment before releasing it to business customers. Production tests are extended to include business continuity/disaster recovery, performance, and security tests.

- *Fix/Repair:*

This step is only to fix issues that are specific to implementation on the production environment.

- *Go-Live:*

Prior to go-live, awareness and training should be provided to all ITSM users. The go-live can be done in several ways to ensure smooth transition with minimum user interruption. The safest approach is the gradual adoption of ITSM processes and modules. Allocating a special task force team of experienced staff to provide support after the go-live is highly recommended.

➢ *Agile Requirement Analysis Process*

It is essential to understand that the Agile Requirement Analysis process works in parallel with other project phases. Its main goal is to optimize the project schedule by analyzing some requirements while implementation of others is ongoing. This process, which can be triggered in the first three phases of the project, has three steps as explained below.

- *Assess Agile Reason:*

This step is to assess the validity of the reason for invoking the Agile process. In this paper, there was a total of 7 common Agile reasons, however, in some environments there may be other different reasons which need to be identified by the ITSM implementation project team.

- *Manage Change:*

This step is the Change Management within the Agile model. In this step, the requirement and its fulfillment actions are assessed against the project schedule, cost, resources, dependencies, alternatives … etc. Once assessed, the requirement can be either approved or rejected. The efficiency of the Agile-Waterfall hybrid model depends heavily on the efficiency of this process. The fact that Agile reasons are identified and pre-agreed on already improves its efficiency. Additionally, during project implementation, the unknowns become less and the risks become manageable, resulting in more efficiency in change assessment.

- *Analyze Requirements:*

Each requirement gets to this step is discussed, analyzed, and prototyped, if needed. Then, the requirement is translated to technical requirement and the project requirement document is updated accordingly. What distinguishes this hybrid model is that after the Agile requirement analysis is completed, the implementation of

the requirement will go back to the Waterfall model, which by nature, has more controlled steps.

V. PRACTICAL RECOMMENDATIONS TO ADOPT AGILE-WATERFALL HYBRID MODEL IN ITSM IMPLEMENTATION PROJECTS

➢ *To Utilize the Agile-Waterfall Hybrid Model for ITSM Implementation Projects, it is Recommended to do the Following:*

- Define ITSM customer functional/ non-functional requirements from business perspectives and prioritize them before choosing the ITSM solution and before engaging the implementation vendor
- After the ITSM solution is selected and an implementation vendor is identified, set an agreement, upfront, between the IT organization and ITSM implementation vendor on using the Agile-Waterfall model and explain its use and how it works. The best place to document this agreement is in the scope of work (or contract).
- Assign a project manager (PM) from the IT organization and another PM from the ITSM implementation vendor. This arrangement is common in projects where every PM has clear responsibilities.
- Build an implementation team from both the IT organization and the ITSM implementation vendor. This team works collaboratively together throughout the entire project.
- Conduct technical sessions during the requirement analysis and design phase. The purpose of these sessions is mainly to document all project requirements and have common understanding and expectations.
- Discuss and agree on criteria for implementation efforts and cost estimation in case of changes happening during the project implementation phase.
- Build and review the project requirement document that has both customer functional requirement and ITSM solution-specific requirement along with information about each requirement. Building a comprehensive project requirement document is a success factor of the project.
- Build the project implementation plan (using project management software) based on the information in the project requirement document. The project implementation plan should have implementation tasks along with start/end dates, resource names, phase…etc. Custom fields can be created in the project implementation plan to include the remaining information (defined in step 1.3) about each requirement such as priority, implementation efforts, implementable, Agile reason…etc. Eventually, both the project requirement document and the project implementation plan are aligned.
- From the project implementation plan, filter the implementable requirements and get an understanding of the volume of requirements that require Agile process. Depending on the volume of Agile process requirements and the size and skills of the project implementation

team, either create two teams to work in parallel in both Waterfall and Agile activities; or continue with implementation with one team and later on create a small team for the Agile requirements.

- When handling Agile Requirement Analysis, it is important to capture and document the Agile reason along with the Agile change management assessment and outcome.
- It is highly recommended to have three ITSM environments: development, test (or QA), and production. This helps greatly in

✓ Prototyping some requirements under Agile Requirement Analysis
✓ Performing partial/unit tests which saves time and efforts

- Arrange two or three weekly meetings to update project progress and track Agile Requirement Analysis
- Manage the remaining activities of the project as a regular project
- Record lessons learned for future reference

VI. BENEFITS OF THE AGILE-WATERFALL HYBRID MODEL

➢ *Below are Benefits of the Agile-Waterfall Hybrid Model Specifically for Managing Requirement Analysis in the ITSM Project:*

- It sets the agreements between the IT organization and the ITSM implementation vendor, prior to the project's start, on how to handle project requirements and their implications throughout the project.
- It clarifies to the ITSM implementation vendor the additional requirements imposed by the ITSM solution itself. It also ensures addressing them properly by giving sufficient time to IT organization to make the best decision instead of accepting quick proposals by ITSM implementation vendor.
- It optimizes the project schedule so that implementation can start while some of the requirements are analyzed in parallel.
- The model optimizes the use of the project allocated budget by exploring and identifying hidden requirements that are related to the ITSM solution itself (ITSM solution-specific requirements). It also caters for new customer requirements by having pre-defined criteria that are used for assessing its implication on cost and schedule.
- The availability of Agile change management ensures more control over the requirements in an efficient way, especially it works after having a better understanding of unknowns and risks.
- The project implementation plan is fully aligned with the project requirement document. Additionally, it is built by both the ITSM implementation vendor and the IT organization which ensures consensus and minimizes rework.
- The model maintains the as-built documentation which is a core requirement for support after go-live.

VII. CONCLUSION

Because of the constraints of the ITSM implementation projects, the Waterfall model is mostly preferred (especially by ITSM implementation vendor) since it is plan-driven and the final deliverables, in the best scenario, are what all parties agreed upon at the beginning. However, the Waterfall model is very inflexible, and eventually the project might not meet all IT organization expectations. As an example, the Waterfall model does not welcome the continuously added features through the rapid deployment of ITSM solutions.

The pure Agile model, on the other hand, is not always practical for implementing purchased software (such as ITSM solutions). This is mainly due to its uncontrolled budget and schedule along with the huge involvement of customers throughout the implementation, besides its openness to scope changes.

The Agile-Waterfall hybrid model, which is common in software projects, optimizes the benefits of both models and can enable achieving IT organization expectations with an optimized budget and schedule. It can be used in ITSM implementation projects to handle the analysis of requirements that either cannot be analyzed easily during the requirement analysis and design phase or that may surface during the implementation phase. To maintain Waterfall controls and Agile flexibilities, the hybrid model defines Agile reasons upfront so that both the ITSM implementation vendor and the IT organization have clear understanding. Similarly, it defines the required duration and cost for new requirements that may surface during implementation with no room for scope creep. Additionally, with the existence of Agile Change Management that works on pre-defined criteria, the Agile-Waterfall hybrid model becomes more controlled and efficient.

REFERENCES

[1]. Reqtest, (2020). Why Bad Quality Of Requirements Will Lead To Project Failure? [online] Available: https://reqtest.com/testing-blog/why-bad-quality-requirements-lead-to-project-failure/#:~:text=If%20 the%20requirements%20of%20a,up%20serving%20i ts%20intended%20purpose.[Accessed 29 Sep 2023]

[2]. Mary Lotz, (2018). Waterfall vs. Agile: Which is the Right Development Methodology for Your Project? [Online] Available: https://www.seguetech.com/ waterfall-vs-agile-methodology [Accessed: 23 Oct 2023]

[3]. Dr. Tuuli Bell (2016). ITIL learnings [online] Available: https://www.bcs.org/articles-opinion-and-research/itil-learnings/ [Accessed 19 Oct 2023]

[4]. Fredrick Dande, (2017). Customization key to successful ITSM implementation [online] Available: https://www.networkworld.com/article/3159994/cust omization-key-to-successful-itsm-implementation. html [Accessed 21 Nov 2023]

[5]. Meghan Corbin. Crossing a Gantt Chart With a Kanban Board: Agile-Waterfall Hybrid in Project Management [Online] Available: https://www.teamly.com/blog/agile-waterfall-hybrid-project-management/ [Accessed: 21 Nov 2023]

[6]. James Freeman. Agile VS Waterfall: A Complete Comparison [Online] Available: https://www.edrawsoft.com/agile-vs-waterfall.html [Accessed: 13 Nov 2023]

[7]. Reka Moksony (2021). When, Why, and How to use the Agile-Waterfall Hybrid Model [Online] Available: https://content.intland.com/blog/agile/ when-why-how-to-use-the-hybrid-model [Accessed: 6 Nov 2023]

[8]. Lucid Content Team. Agile-Waterfall Hybrid: Is It Right for Your Team? [Online] Available: https://www.lucidchart.com/blog/is-agile-waterfall-hybrid-right-for-your-team [Accessed 29 Sep 2023]