35241 Drakeshire Ln

Farmington, MI, 48335.

6/14/2020

[dgaonkar@unomaha.edu](mailto:dgaonkar@unomaha.edu)

402-590-3286

Mr. Davis, Professor

University of Nebraska Omaha

6001 Dodge St

Omaha, NE 68132.

Dear Mr. Davis,

Here is my final report on the Software Development Methodology. This report includes a title page, table of contents, executive summary, and the main context of the report. At the end of the report is a working reference list.

The start of my report gives a brief introduction to how software is developed, describes why software fails then it goes on explaining the software development life cycle that is a crucial factor while developing software. Later we look at various methodologies, advantages, and disadvantages, the software risk associated and then a combination of two methodologies is presented. I end my report by describing the software testing and maintenance methods.

I hope the information that I have provided in my report is interesting and easy to read. My report will educate both the IT professionals and the audience about software development methodology. By reading this report, the audience will gain knowledge on this topic and apply it to business according to their organization's needs.

Sincerely,

Diksha Gaonkar.

UNO Management Information Systems Major

**Software Development Methodologies.**

For

Thom Davis

Technical Writing Instructor

University of Nebraska – Omaha

Omaha, Nebraska

By

Diksha Gaonkar

CIST – 3000 Student

July 1, 2020

**Table of Contents.**

Executive Summary ……………………………………………………………………...... iii

Introduction………………………………………………………………………………… 1

Brief Definitions…………………………………………………………………………….1

Why Software Projects Fail? ……………………………………………………………….1 - 2

Software Development Life Cycle Stages…………………………………………………………………………….….………3

Software Development Methodologies………………………………………………….….4

Predictive……………………………………………………………………………4 - 6

Iterative……………………………………………………………………………...6 - 7

Adaptive…………………………………………………………………………….7 - 8

Software Development Risks……………………………………………………………….9

Predictive Method or Adaptive Method for Software Projects…………………………….10

Do Agile Methods Work for Large Scale Projects? ……………………………………….11

Hybrid Development Methodologies……………………………………………………….12 - 13

How to manage Agile and Waterfall Together………………………………...........13

Software Testing…………………………………………………………………………….14

Software Maintenance……………………………………………………………………….15

Conclusion……………………………………………………………………………………16

References………………………………………………………………………………….17 - 18

**List of Figures and Table.**

Table 1: Agile Development Method per budget size category………………………………11

Figure 1: A general model of software development. (Author Created) …………………….1

Figure 2: Average percentage of projects completed within a specific timeframe…………...2

Figure 3: Displays the percentage of usage of methodologies in industries………………….4

Figure 4: Waterfall Model Development Process…………………………………………….5

Figure 5: Iterative Development Process………………………………………………….….6

Figure 6: Agile Methodology Development Process…………………………………………7

Figure 7: Hybrid Development Methodology Process……………………………………….12

Figure 8: Black Box Testing Method…………………………………………………………14

Figure 9: White Box Testing Method…………………………………………………………14

Figure 10: Maintenance Process in an Organization………………………………………….15

**Executive Summary**

Developing software is a very complex and time-consuming process that requires communication with the client and the project manager. Before developing software, the client requirements are acquired, and the software development starts. Sometimes in between the development process the client requirements changes due to which the scope of the project changes which leads to the increase in the budget and this sometimes might lead to project failure. Time and cost are the biggest development challenges.

The Project Management Institute (PMI) believes that a software/ project development methodology is critical to business performance and the organization's success. This report explains the software development life cycle where first the requirements are gathered, through research and planning is done, then we design the system and obtained the client's feedback once it approved we move to the next phases of the life cycle which are the development of the software. Later it tested for errors and then is implemented in the organization.

There is three main software development project life cycle: Predictive also known as a waterfall (client requirements are obtained first; the product is developed and then feedback is obtained), Iterative Model (works with few initial client requirements; software is developed and provides to the client for feedback; new requirements are obtained and then the product is released to the client for further requirements and the cycle continues until all the requirements are met), Adaptive Methodology is also known as the Agile Methodology(works with few client requirements, divides tasks into smaller sprints, development is carried out in one sprint cycle and then testing is done, new client requirement can be obtained then another sprint cycle is carried out and testing is done and the cycle is done. It is like the iterative model but agile provides more flexibility)

The next section of the report says which methodology (either Agile or Waterfall) can be selected based on the nature of the project, requirements, time, budget, organizational process, and the involvement of the product owner. A brief description of risk is included in this report. It is very important to know this risk when selecting a methodology for a project. The Agile and Waterfall Methodologies are combined to obtain a hybrid methodology. Due to the complexity of the projects, it is often beneficial to use a hybrid methodology as the strength of two methodologies will be applied while developing and it will minimize the weakness and the risk for the project.

The last section of the report describes the software testing and software maintenance. Once the software is developed it needs to test and implement. A testing method can allow us to find if any errors exist in the system before it is implemented in the organization. Testing is carried out with two methods either a black box test or a white box test. Once the software is implemented, we need to have proper maintenance procedures in place so when any unforeseen circumstances occur the system does not fail.

**Introduction**

The process of the development of software is a very complex task that requires continuous communications with the client and the project managers. The primary objective of any software development project is to create a high-quality maintainable software within a reasonable amount of time, and it should be within an affordable budget. This is only achievable if we can determine in advance how much time and effort is going to be required the production software. Software development organizations need to follow several methodologies while they are developing the project. "Software Development Methodology is a framework used to structure, plan, and control the processes of developing an information system."(Source: Google) Choosing the right methodology for an IT project implementation is a high impact decision that affects the business goals, response time, and the end-user product.

Implementation

Design

Testing

Planning

*Figure 1: A general model of software development. (Author Created)*

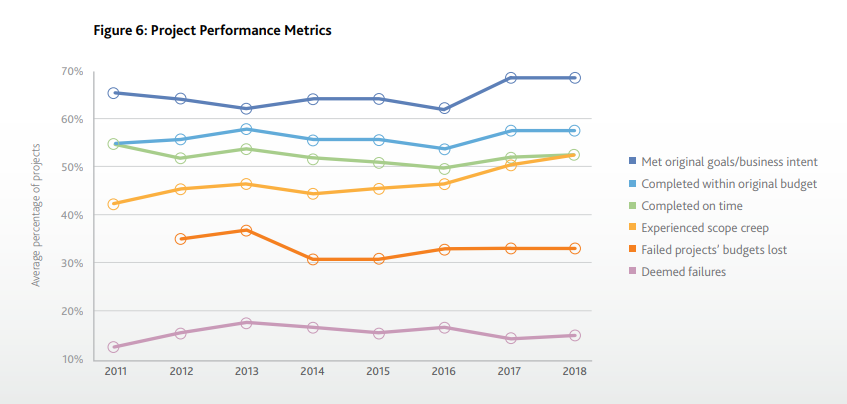
**Definitions**

In this report, the software development methodology and the project development methodology are considered as the same. The business requirement documentation is a document that describes the high-level business needs obtained from the clients. The function requirement documentation is a document that outlines the functions required to fulfill those business needs. The IT Product and IT project are the same. The Production Environment is where software is put into operations to produce the end-user application.

**Why Software Projects Fail?**

In the early days of computing, software development processes were not complex, and a single person could easily handle them. The project that was supposed to be developed was first clearly understood and then developed. There was no distinction between the client requirements and the end-user application. This model was neither controlled nor formulated. When the project complexity increased this model caused many difficulties as no documentation was created, no business rules were followed. To produce a high-quality product, we should understand business requirements to develop use-case analyses and design specifications. This is a lengthy process that the client may sometimes struggle to see or understand. They say that they want a certain product and it is up to developers to figure it out on how they can develop. Sometimes initial requirements change due to which developer needs to start the process again from scratch.

Developing a product without the guidance of a development methodology often leads to a project which is delivered late or over budget or even some time to complete project failure. Less than one-third of all projects were completed on time and in the budget in 2016. "For one billion invested in the United States, 122 million was wasted due to the lack of project performance. 97% of organizations believe that project management methodologies are critical to business performance and organizational success. 80% of the executives do not know how their projects align with their company's business strategy." (Burger, R. 2016) Businesses have identified time and cost as their biggest development and management challenge. Some of the businesses lack the skills and resources to manage the project efficiently. Adhering to a development methodology will allow the developers and the business analyst to provide better estimates which will keep our client informed about the progress of their project. It will also help in providing a clear understanding of tasks ahead of time. This gives us enough time to understand the risk allowing time to adjust and the result will provide a stable system to the clients.



*Figure 2: Average percentage of projects completed within a specific timeframe (.2016)*

**Software Development Life Cycle Stages.**

Developing a software project is a process that consists of various distinct stages. Every stage has its deliverables and needs to be completed within a specific time for the implementation within the organization.

1. Research: This stage involves the project owner (the sponsor) and the project team. The project team gathers and exchanges the client requirements within their teams. To properly formulate the client requirements a proposal is created which defines the set of goals, documents the cost-benefit analysis, and the risk management plan. Requirements are evaluated from both the business and the technical perspectives. Once the project is approved the team members research for frameworks, APIs, and hosting infrastructures that are required to build the software application.

2. Planning: This stage defines the entire flow of the application. A project management plan is developed. The project's tasks are divided into easier, manageable subtasks. The planning phase involves the project team members to decide a software development methodology and a work protocol that suits their project.

3. Design: This stage transforms the client requirements into a design for the application. The design can be displayed to the owner as the preview of the application that is going to be built. Sometimes in this stage, the project owner comes up with new requirements.

4. Development: This stage converts the design into an information system. The development stage sets up the development and testing environment. These environments are synchronized with the same protocol. The application code is written in the development environment and then it is uploaded onto the testing environment. The project manager determines the actual progress and evaluates it against the initial planning.

5. Testing: This stage demonstrates the developed application as per the client requirements defined in the Functional Requirements documentation developed by Business Analyst. This stage produces the test analysis report.

6. Implementation: This stage involves the production of the application in the organization. Once the application is developed and completely tested against all the test cases the system is forwarded into the production environment. Back-up procedures are defined and tested.

7. Maintenance: This stage describes the tasks and the operations to maintain the information system in the production environment. Sometimes issues are not identified in the testing and the implementation stages that is why systematic testing is performed in this stage so that all are the issues are resolved. It involves post-implementation and in-process reviews.

**Software Development Methodologies.**

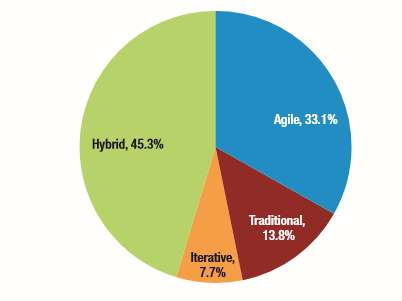
"A software development methodology is a set of rules and guidelines that are used in the process of researching, planning, designing, developing, testing, and maintaining a software product." (Vijayasarathy et al., 2016)

The three main IT project lifecycles are

1.Predictive. (Traditional)

2. Iterative.

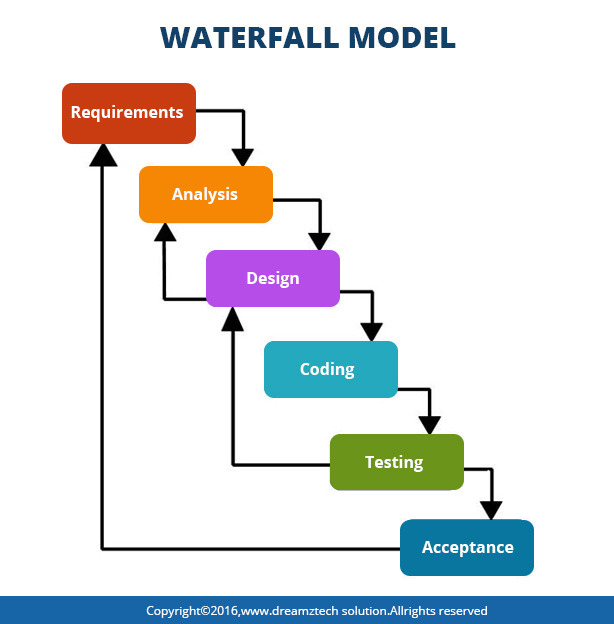
3. Adaptive. (Agile)



*Figure 3: Displays the percentage of usage of methodologies in industries.* (Vijayasarathy et al., 2016)

**Predictive Methodology**

The predictive methodology is also known as the waterfall methodology. It is a software development process that was first introduced in the year 1970 when software projects started requiring teamwork. "Waterfall methodology is a linear project management approach where the stakeholder and client requirements are gathered at the beginning of the project and then a sequential project plan is created to accommodate those requirements."(Source: Google) It is one of the oldest models after Cowboy methodology. “This methodology was adopted in various industries due to its ease in implementation. It was widely used in many government projects.” (Jones, 2017, pp. 523)



*Figure 4: Waterfall Model Development Process. (Source: Google)*

The waterfall model focuses more on the quality of the system. The stages are followed one after another. There is no overlap between any stages. Each stage has its deliverable. “It is a very predictable model that values rigorous software planning and architecture. The project owner can provide their feedback only when the software application is completely developed and tested properly. This methodology is mainly suitable for software projects where the requirements are clear, and detailed planning could be done before starting the project.” (Jones, 2017, pp. 523- 524)

"Waterfall development has been used in more than 2,000,000 projects between the year of 1970 and 2014 and some major applications still use it." (Jones, 2017, pp. 523)

Examples where waterfall model is used:

1. SAP.
2. Oracle.
3. IRS taxation packages.

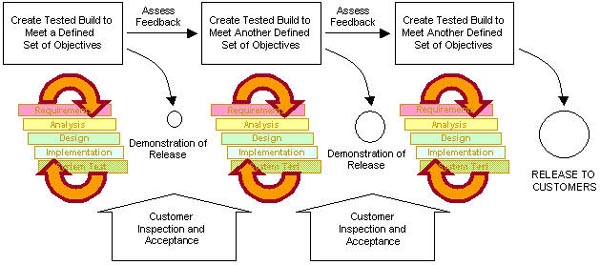
The main advantages of the waterfall model.

1. Client requirements are already approved and documented.
2. Easier for project managers to provide accurate estimations of the project cost, resources that will be required.
3. Accurate measurements of the progress as we move through the different phases.
4. Since new requirements are not added during any phases production delay will not occur.
5. Waterfall methodology is widely known in theory and used in industries.

The disadvantage of the waterfall model is that it tries to accomplish too much too soon, such as the Requirements need to be completely developed before starting the design. In today's world due to the dynamic changes the waterfall fall phases are not completed before the next phase begins. "Requirements are usually about 50% completed when design starts; the design is only about 60% complete when coding starts; coding is only about 35% complete when testing starts. These overlapping phases make project planning difficult." (Source: Google)

**Iterative Development.**

One of the biggest problems of waterfall methodology is that sometimes client requirements change due to which developers need to start again from the very time-consuming planning phase, and it causes the scope, cost of the project to increase. This creates a demand for a new methodology that could provide faster results, can create software that takes less up-front requirements, and can offer greater flexibility. "Iterative development is a way of breaking down the software development of a large application into smaller chunks."(Source: Google) That is the model attempts to gather the full specifications of requirements and focuses on the simplified user features and then develops a more complex set of features until the end-user application is completed. With the help of Iterative Development, the developers can demonstrate the end-user application to the client, obtain valuable feedback from the client, and apply that feedback in the next iterative cycle.



*Figure 5: Iterative Development Process. (Source: Google)*

Every iteration is a small-waterfall process which involves a feedback mechanism from one phase to another. It provides design information for the next phase. Every iteration lasts for about two to six weeks. The software product that is produced at the end of the iterative cycle will be checked against the client requirements mentioned in the functional requirement document. Once it is approved it can directly go into the production cycle and can be implemented into the organization.

Iterative development is useful for projects where client requirements are likely to change. Iterative development was used by NASA space shuttle in the year of 1977 and 1980. It is still widely used in defense systems and software packages.

The main advantages of iterative development model.

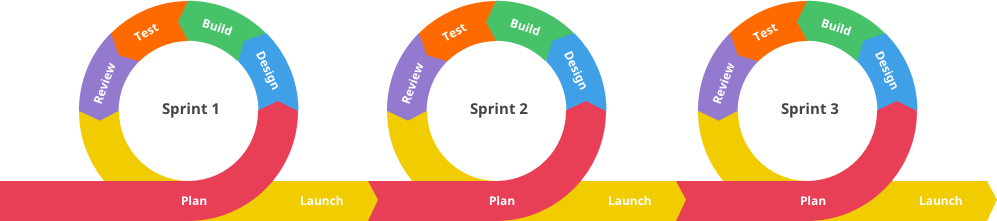
1. Projects are developed quickly at the beginning of the development cycle.
2. Projects can be parallelly developed and progress is easily measured by comparing it to the initial requirements.
3. Risks identified in the first iteration cycle can be resolved in the next iteration cycle which provides flexibility for the changes in the client requirements.

The disadvantages of the iterative development model.

1. Requires more resources than the predictive methodology.
2. Takes a long period for project completion since all requirements are not available upfront
3. It can cause design issues.
4. This model is not suitable for small scale project as the project are difficult to manage.
5. Risk analysis might require the involvement of highly qualified specialists which increases the project budget.

**Adaptive**

The adaptive methodology is also known as the agile methodology. It an evolutionary model that is based on incremental and iterative development. The agile methodology is an alternative to the traditional model. "Agile methodology is a type of project management process, mainly used for software development where demands and solutions evolve through a collaborative effort of self-organizing and cross-functional teams and their customer."(Source: Google) In agile the developer team can generate and respond to changes that occur in uncertain environments.



*Figure 6: Agile Methodology Development Process. (Source: Google)*

The software development process is carried out in small cycles also known as sprints. A sprint can be finished with two-three weeks and then it undergoes a thorough testing process. Once it is thoroughly tested, they provide the client with the system functionality. The client's feedback is required after each sprint is finished. If the client approves the system, it goes through the production unit. This results in an incremental release with each release been built over the previous functionality. The entire projects take over multiple sprints. "Requirements and solutions are evolved through collaboration between self-organizing cross-functional teams."(Source: Google)

Agile development methodology has better quality and productivity as compared to the older waterfall model.

The advantages of Agile Methodology.

1. High risks are not involved due to the flexible process change.
2. The system versions are released at a faster rate.
3. Projects are developed into a shorter sprint.

The disadvantages of Agile Methodology.

1. The existing requirements may conflict with the new requirements.
2. Code reverting might occur at a frequent rate.
3. All team members need to be highly professional and client oriented.
4. The projects might exceed the expected time.

**Software Development Risk**

The process of developing software involves specific requirements that can accompany various stages of the software development life cycle. By selecting a specific methodology, the occurrence of the risk could vary. “The term software risk refers to unexpected events that may cause a negative impact on the software product, the duration needed for its development, and the resources used.” (Georgiev & Stefanova, 2014) Risks are identified in two categories: external risk and internal risk.

1. External Risk: These risks are extremely difficult to handle. At the project manager level, these risks could be identified. To manage these risks the senior management needs to be involved so proper measures could be taken.

Example. Rapid application development can cause a change in direction.

1. Cost Risk: These types of risks are either directly or indirectly under the product manager’s control. It could increase the original estimated budget allocated for a software project.

Example. Overrun of budget and schedule.

1. Schedule Risk: These are the types of risk that could lead to massive project failure or it could delay an opportunity for the product.

Example. Inaccurate estimation during the early stages of development cycles could lead to schedule risks.

1. Technology Risk: These risks could occur when a project fails to meet specific systems functionality or the performance expectations from the clients.

Example. Problems with using the wrong tools, failure to understand the complexity of the project, developing an incorrect API.

1. Operational Risk: Operational risk is identified as the inability of implementing a large-scale project change efficiently. These risks could result in a project failure or the expected benefits.

Example. Lack of collaboration between the teams, implement the project too soon.

The software development team needs to choose the right methodology for the software development system to avoid these risks and their mitigation. There is numerous methodology that is available that can be applied to the development of the software but the most widely used methodology is the Agile methodology. It is the most productive and the most suitable methodology.

**Predictive Method or Adaptive Method for Software Projects**

Several factors play a role while deciding which methodology works best for developing software projects.

1. Requirements

Waterfall: Waterfall Software Development Methodology can be applied when a project

has firm regulatory requirements and very little space for changes.

Agile: Agile Development Methodology does not require firm regulatory requirements and the project can be started even with few initial requirements.

1. Nature of the Project

Waterfall: If the features and the interface of a system are already defined and if the team is working on the existing system for the upgrades then the Waterfall methodology is suitable.

Agile: If an organization is building a new system then they should use Agile development methodology as it helps to discover the functionalities of the projects iteratively.

1. Involvement of the Product Owner

Waterfall: If the project has defined the requirements in the earlier phase then waterfall methodology does not require much involvement from the product owner. The Product owner could provide feedbacks during project check-ins.

Agile: For the agile development methodology the product owner needs to be deeply involved as they are the member of the team and the owner of the product. They are responsible for making decisions about the scope, and the features of the product.

1. Organizational Processes

Waterfall: If an organization follow a set of sequence for developing a product which involves inflexible processes then waterfall methodology is best suitable.

Agile: If an organization allows working flexibly and does not have any precise processes to follow for product development then agile methodology would be beneficial.

1. Timeline

Waterfall: If the timeline for a project is fixed then waterfall development methodology offers a higher success completion rate and the end-product provides the necessary functionalities.

Agile: If the project needs to be delivered within a short period then Agile is an appropriate method as Agile Methodology focuses on developing a product rather the documenting the requirements and the processes involved.

1. Budget

Waterfall: Works best when the project budget is fixed. Provides accurate outcomes.

Agile: Agile methodology focuses on features and speed rather than the cost. Sometimes ideas could turn up and to assemble those ideas a team would require more time and budget.

**Do Agile Methodology work for Large scale Software Projects?**

Agile Methodology helps an organization to improve its performance and to provide an end-product within a shorter period. Originally Agile methodology was designed to be used in small scale and individual projects but as the benefits increased organization started applying agile methods in large scale projects. “Project success was measured as the combined performance of the project regarding the client benefits, cost control, and time control. In March 2015, a question-based survey was conducted in Oslo, Norway to analyze information on 101 Norwegian software projects.” (MJ Jørgensen, 2018)

Some of the questions included were:

1. Describe how agile the projects were.
2. The project’s success and budget
3. The project outcomes in terms of client benefits cost control and time control.

“The projects were ranked from 1 (very agile) to 5 (not agile). In this survey the participants responded as 17% that their projects were very agile; 40% as partly agile and 4 % as not agile.

37% of the projects were successful, 32% were acceptable and 31% were either not completed or failed.” (MJ Jørgensen, 2018)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Budget Size | Agile | Partly Agile | Not Agile | Number of projects |
| Small | 37% (18) | 42% (20) | 21% (10) | 48 |
| Medium | 58% (15) | 19% (5) | 23% (6) | 26 |
| Large | 33% (9) | 56% (15) | 11% (3) | 27 |
| Number of projects | 42 | 40 | 19 | 101 |

*Table 1. Agile development method per budget size category. (MJ Jørgensen, 2018)*

“The participant's response to agile methods in terms of client benefits cost control and time control.

Client Benefits: 35% of projects were successful; 55% of projects were acceptable and 10% failed.

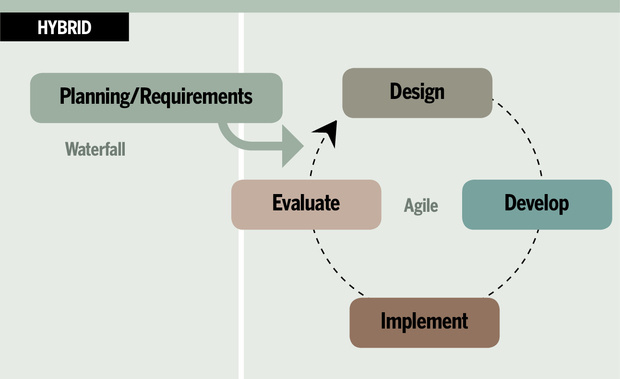
Cost Control: 30% of projects were successful; 32% of projects were acceptable and 38% failed.

Time Control: 37% of projects were successful; 32% of projects were acceptable and 31% failed.” (MJ Jørgensen, 2018)

Large scale projects are not easy to implement and they are often at risk of failing because of the lack of clarity of scope, collaboration within the team, and relying on legacy systems. The result from the survey shows that the non-agile method provided higher success rate in the small and medium scale projects. The agile methods and partly agile methods work well on a large-scale project. As the size of the project increases it is preferable to adopt to agile methodologies.

**Hybrid Development Methodologies**

A possible cause of software project failure could be an improper choice of a methodology. The main objective of hybrid methodologies is to maximize the benefits, strength of the two methodologies and to reduce the weaknesses to improve the development cycle. It can produce high-quality products. The waterfall is a predictive methodology that was first used in the late 1960s. Agile is a newer methodology released in February 2001. We can mix both waterfall and agile methodology to obtain a hybrid methodology. It works well and offers several advantages as compared to waterfall and agile alone. “Project Management Institute found that project managers used the hybrid method in 23% of projects.” (Ozhiganov, 2019)



*Figure 7. Hybrid Development Methodology Process (“Source: Google”)*

The hybrid approach helps the project manager to choose an appropriate methodology for each phase such as waterfall methodology could be applied to gather the client requirements and designing, developing, and testing could be carried out with the Agile Methodology. This combination provides the flexibility of Agile along with rigorous planning, estimating, and quality control. The hybrid methodology is best suited for large and complex projects which require to meet strict deadlines and develop the new functionalities frequently. The model is best suited for projects that are supposed to be delivered in a specific time frame but have changing client requirements.

Some Advantages of Hybrid Methodology:

1. The development team and client agree on every deliverable output at an earlier stage in the systems development cycle.
2. Since the entire scope is already planned out it is easier to compare the progress of tasks.
3. Test cases and the code development could be done simultaneously from the requirements documentation.
4. The business analyst could document all the client requirements in the requirements document.

Some Disadvantages of Hybrid Methodology:

1. Continuous administrative intervention to resolve team conflicts which can result in a waste of time and effort.
2. With the hybrid methodology, it is difficult to present the project funding.

**How to manage Agile and Waterfall Together.**

Different organizations follow different approaches when they are developing a software product. A single methodology cannot be applied to all software projects. A hybrid methodology can work well for projects as it is utilizing both traditional and adaptive methods. The hybrid methodology can allow the team to define the requirements earlier in the planning phase and then adapt to the changing requirements through feedback and delivery mechanism. With the waterfall model, we can track the progress of the product development along with the adaptability and the flexibility obtained from the agile methodology.

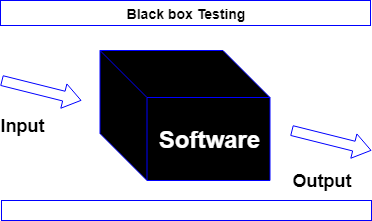
The development team needs to have a clear understanding of how the hybrid methodology works and how it can implementation otherwise the entire project could lead to failure. Collaboration and cooperation from the team members and the client is the key to success. Some of the best practices of managing agile and waterfall model together are:

1. The client expectation should be documented clearly at the start and the end of every sprint cycle and can ask the client for a signoff for each task delivered at the end of the sprint.
2. If a task is not specified properly or if the client requirements change the business analyst team can issue a change order and the change order should include both schedule and cost impact.
3. A task review meeting should be conducted for every task.
4. The project progress needs to be analyzed regularly and evaluate how effective the new processes are and where changes might be required.

**Software Testing**

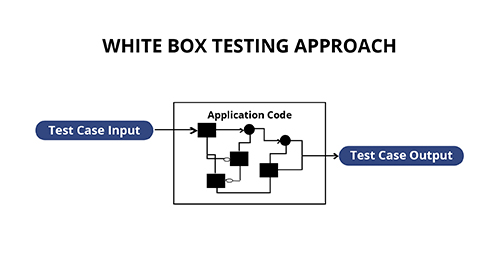
After developing software, testing needs to be carried out to detect errors. The software testing method allows the software team to demonstrate that their software works well in all the conditions even under exceptional situations. The testing can allow the business analyst to verify if the system is accurately designed and developed according to the requirements documentation. It can provide information on the quality of the product. Testing can be carried out anytime during the software development life cycle and the testing model needs to be associated with the appropriate methodology selected for the software development.

The software testing methods are divided into two categories: white box testing and black-box testing.



*Figure 8. Black Box Testing Method (“Source: Google”)*

In the black box testing, the software is tested without knowing the implementation. We check whether the expected output and the actual output are similar if the results are similar then the software is developed accurately. This black box testing method is usually applied to the higher level of tests such as acceptance testing.



*Figure 9. White Box Testing Method (“Source: Google”)*

In the white box testing, we know the internal mechanism such as the algorithm that is applied while testing. The white box testing method is applied to unit tests and on the integration tests.

**Software Maintenance**

Once the software product is developed and accurately tested the software product needs to be properly maintained so that any faults or errors generated could be fixed. The developed software products need to adapt to the environment changes. “IEEE standard has defined maintenance as: the process of modifying a software system or component after delivery to correct faults, improve performance or other attributes or adapt to a changing environment. The maintenance phase in the software development life cycle is the most expensive and time-consuming stage.” (ZIMA, 2015)

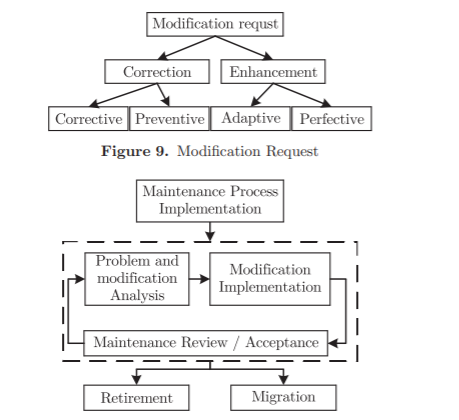
The maintenance process could be divided into four categories:

1.Adaptive: This is related to the changes in the environment.

2. Perfective: After the software is released functional changes could be done, performance tuning.

3. Corrective: To fix all the errors and faults.

4. Preventive: Take proper measure for better software maintainability.



*Figure 10. Maintenance Process in an Organization* (ZIMA, 2015).

Adoption of proper maintenance practices by a software organization during every phase of the software development life cycle could reduce the number of failures once the system is delivered.

**Conclusion.**

In this report, the three main methodologies are described as the waterfall, agile, and hybrid methodologies. This paper also describes how the methodology is carried out while developing software, their advantages, and disadvantages. The risk for software development processes is identified. Depending upon a project or the application a methodology needs to be selected, before choosing a methodology an analysis needs to be done on the team member expertise, the complexity of the project and the schedule, and the cost-benefit analysis.

Every organization needs a software product that is developed perfectly and work in all the scenario but in the real-time methodologies cannot fulfill all the requirements. To select the best and the appropriate methodology the organization needs to consider all the factors. Collaboration is very important while developing a software product. With the help of the experienced team members, project managers, business analysts a particular methodology could be adopted. Sometimes selecting one methodology does not provide an organization with all the expected benefits it is always better to select a combination of methodologies for the development process to produce a better outcome.

Flexible methodology and their practices in the software development process provides the development team a privilege to connect with the client and the product owners. The proper understanding of the client requirements could lead to the improvement of the quality of the end-product by minimizing the risk and the number of errors in the software. The incremental development approach and continuous communication with the project manager could reduce the schedule, cost, and technological risks.

Reference List

Abdalhamid, S., & Mishra, A. (2017). Adopting of Agile methods in Software Development Organizations: Systematic Mapping. *TEM Journal*, 6(4), 817-825. Doi: 10.18421/TEM64-22.

Bilgaiyan, S., Sagnika, S., Mishra, S., & Das, M. (2017). A Systematic Review on Software Cost Estimation in Agile Software Development. *Journal of Engineering Science and Technology*, 10(4), 51-64. Doi: 10.25103/jestr.104.08.

Burger, R. (2016, September 26). *20 Surprising Project Management Statistics*. Retrieved June 14, 2020, from https://blog.capterra.com/surprising-project-management-statistics/

Elgabry, O. (2017, March 17). Software Engineering — *Software Process and Software Process Models (Part 2)*. Retrieved June 14, 2020, from https://medium.com/omarelgabrys-blog/software-engineering-software-process-and-software-process-models-part-2-4a9d06213fdc.

Francino, Y. (2019, September 11). *How to manage agile and waterfall together*. TechBeacon. Retrieved June 3, 2020, from https://techbeacon.com/app-dev-testing/managing-agile-waterfall-together.

Georgiev, V., & Stefanova, K. (2014). *Software Development Methodologies for Reducing Project Risks*. Semanticscholoar. https://pdfs.semanticscholar.org/966f/2c37bdb24259b9482198177b042d9533d0cd.pdf.

Jones, C. (2017). *Software Methodologies* (1st ed.). Taylor & Francis.

Jørgensen, Magne. (2018). Do Agile Methods Work for Large Software Projects? *Lecture Notes in Business Information Processing*, *314*, 179–190. https://doi.org/10.1007/978-3-319-91602-6\_12.

Maric, M. (2019, January 29). *How to select the best project management methodology for your IT business*. ITProPortal. Retrieved June 3, 2020, from https://www.itproportal.com/features/how-to-select-the-best-project-management-methodology-for-your-it-business/

Ozhiganov, I. (2019, August 9). *Why and How to Use a Hybrid Project Management Approach*. Medium. https://medium.com/azoft/why-and-how-to-use-a-hybrid-project-management-approach-4d58b45e9a84#:%7E:text=Project%20Management%20Institute%20found%20that,method%20in%2023%25%20of%20projects.&text=The%20point%20of%20the%20hybrid,methodology%20for%20each%20project%20phase.

Vijayasarathy, L., & Butler, C. (2016, September), THE IEEE COMPUTER SOCIETY. *Choice of Software Development Methodologies*. THE SOFTWARE ENGINEERING PROCESS.http://www.sc.ehu.es/jiwpiacj/Informatika/2016\_\_Choice%20of%20Software%20Development%20Methodologies%20Do%20Organizational,%20Project,%20and%20Team%20Characteristics%20Matter.pdf.

ZIMA, D. (2015, October 1). *MODERN METHODS OF SOFTWARE DEVELOPMENT*. TASK QUARTERLY. https://task.gda.pl/files/quart/TQ2015/04/tq419v-c.pdf.