Multi-Dimensional Success Factor of Agile Software Development Project.

Mahiul Alam Sad
Department of Computer Science & Engineering.
American International University Bangladesh.
Email: mahiulalamsaad@gmail.com

Shionty Ghosh
Department of Computer Science & Engineering.
American International University Bangladesh
Email: shiontyghoshdola77@gmail.com

Punam Ghosh
Department of Computer Science &Engineering.
American International University Bangladesh.
Email: punamghosh52@gmail.com

Fahim Faisal
Department of Computer Science & Engineering.
American International University Bangladesh
Email: fahimfaisal562@gmail.com

Dr. Dip Nandi
Professor, Director
Department of cs faculty of Science and technology.
American International University Bangladesh
Email: dip.nandi@aiub.edu

Abstract:
This study aims to research the multi-dimensional success determinants of agile software development initiatives. Although using agile approaches to deliver software projects has gained popularity, ensuring their success still poses substantial challenges. This study attempts to identify and comprehend the essential success elements that influence the success of agile software development projects by a thorough assessment of the literature and case studies of successful agile projects.

The study contributes to the understanding of how organizations can effectively implement agile methodologies and improve the success rate of their projects. The results of this research have implications for both academia and industry, providing a comprehensive guide for organizations to achieve successful outcomes in their agile software development projects. The findings of this thesis will offer valuable insights into the multidimensional success factors of agile software development projects and contribute to the advancement of knowledge in this area.

Index terms: Agile software development, Success factors, Multidimensionality, Project success, Software project management, Agile methodologies, Critical success factors, Project delivery, Software development processes, Agile project management.

1. Introduction:

Agile software development is an iterative, adaptable method of creating software that places a focus on teamwork, regular feedback, and continuous improvement. Since companies look to quickly and effectively respond to customer requirements and market demands, this method has grown in popularity in recent years. While there is no single definition of success in an agile software development project, several factors are commonly considered to be critical for the success of these projects. These success factors can be broadly categorized as technical, organizational, and people-related, and they can interact and influence one another in complex and dynamic ways.

The technical success factors of an agile software development project are typically related to the quality, functionality, and performance of the software product being developed. This includes factors such as the ability to meet customer requirements, the ability to meet performance targets, and the ability to meet technical standards and best practices. Technical success factors are also closely tied to project management, as effective project management practices can help ensure that these factors are met.
Organizational success factors are related to the culture, governance, and structure of the organization, as well as the alignment between the goals of the organization and the goals of the agile software development project. This includes factors such as the ability to align stakeholders, the ability to foster a culture of collaboration, and the ability to provide effective leadership and governance.
People-related success factors are related to the skills, experience, and motivation of the individuals involved in the agile software development project. This includes factors such as the ability to attract and retain high-performing individuals, the ability to build and maintain strong and effective teams, and the ability to foster an environment of trust and collaboration.

In order to maximize the chances of success in an agile software development project, it is more important to consider all of these factors to understand how they interact and influence one another. This thesis will examine the multi-dimensional success factors of agile software development projects in greater detail way to explore how these factors interact and influence one another also how organizations can manage these factors to maximize the chances of success in their projects.

2. Literature Review:

SDLC(Software Development Life Cycle) is very much important for developing software. This is one kind of process in that the clients are well understood about their project and how it will be increased and the overall development process. There is another measurement like is the project process on right track or not. In SDLC there are 5 phases. (a)Analysis (b)Design (c)Implementation (d)Testing (e)Evaluation. After collecting the whole requirement from the client the requirement collector will discuss it with the client & will explain that it is understood. After that design has to be started. After designing development team has to call a meeting with the client, have to show them that all requirements have well understood. Then after that, if the client thinks that there

should change a litter then have to change it for customer satisfaction and well done the project. SDLC is one of the theoretical concepts that, every day, worldwide all IT companies are using this concept and working with this process.[1]
There are so many types of the software development life cycle. Some of them are given below.

Agile Development Model: Agile development methodology is based on iterative and evolutions method combination. Agility is the authority to make or create and respond to change in order to gain profit in business condition. Agile method is mainly iterative and incremental development process [18]. Agile method follow some phases or stage to complete software projects. Requirements collecting is one of the most important part to kick start agile processing. After that design the requirement how to start iteration or construction. In the Development phase time all the development team work how the software more user friendly and develop consumers work environment. When Successfully develop software most crucial time is testing or quality assure time. Most beneficial part of this development stage is client are free to add or reduce their requirement after every iteration. When client give them feedback about their product project member can fix all those problem or add new additional requirement properly.
Waterfall: Waterfall model is one of the most oldest model in SDLC. This model maintain a sequence. Here every steps will be stared before properly done it's previous steps. There have some steps like 1.Analysis & requirement collecting, 2.Design 3.Development, 4.Testing 5.Implementation 6. Maintenance . In this development process each phase will work at it's own time because overlapping is not allow here. After all types of analysis & requirements should be clearly collected then have to go for another steps. After that the design phase will be started. Then after totally complete the design phase then have to start coding phase. Then when coding and implementation will be totally done that time have to start the next phase testing phase will be started. At the same process every step will be done after completely done previous process. Testing of the software is done at the last of the development . That time is to very difficult to modify the software if there have any types of fault arise. So as defects are found at the last so, this is one of the disadvantage of this development process. As we know that all phases work is done not at same time and the rule is after completing one step then you can go the another steps that's why tester can not do anything at the first. If we can follow at the steps of this development process, then we can see that the testing phase is at the last and there have know role of tester before the testing phase. So all types of requirements will be fully clarify at taking requirement from clients and have to done all phase very carefully because as testing phase is last phase that's why it is so difficult to modify at last for the developer after getting report from tester. It is called linear model that's why it is easy to implementation. Documentation is mandatory for each phase at this development. This is good development process for a small project where all types of requirements are very clear.[2] Besides there have many risk after using this model. This model is not good for a large and big budget project.[4]

V-model(Verification & Validation Model): This Verification & Validation Model is better & improve version of the waterfall model in software development process. Here all software development steps are as same as waterfall model like 1.Analysis & requirement collecting, 2.Design 3.Development, 4.Testing 5.Implementation 6. Maintenance . But at every phase here need validation and verification test is mandatory . In every phase you have to check validation and verification of the work otherwise can't move forward or other step. As in every steps here checking validation and also performing verification that's why here testing team and developers work together. This model has of the good side is that each phase is here deliverable. Besides another advantage is that requirement change is possible in any phase besides as in every phases there have a testing part that's why all phases is almost clear without any problem. But besides the advantage there have big disadvantage that, if there have arise any kind of requirement change that time you have to change both requirements documents and test documents[2]. On the other side ,there have no clear path and clear prototype for this development process.[3].This development process is not good for any short term project as there have a testing in every phase of this development that's why it is not possible at a short time. This model is very easy to apply. [2]

Rapid Application Development Model: this model is mainly combination of prototyping and iterative. By using rad model , it is easy to in crease development time. Because development time is very rapid. Here it is possible to reuse the features that help to reduce time.[6]. Cost of this model is low budget and light weighted process. Very easy to change requirements when using this model. Besides maintenance is very easy. Project time is short. Risk is very low. Need users involvement at the beginning of the model.[7] But there have some disadvantages like when this model is using at any big project that time in the project team need high skilled

engineers.[6]. there have overlap of phases in this development process.by using this model , success rate is high. After completing the coding part of the project then testing started. team size is small[8]

Spiral Model: this spiral model was developed by using idea of waterfall model and iterative model.at the primary stage of the process have to collect three types of requirements like system requirements, sub system requirements and unit requirements. In this process model clients will be also very active. Then have to start the design phase. In this phase developer team have to design of the module, architecture of the software, physical out put of the product and besides overall final design. Then have to come the development phase or construction phase and here developer do all logical operations. After that comes evolution and risk analysis started. Mainly this model is used in various industry and it is very popular at industry. This process model is good for medium type projects. Here very important thing is risk analysis . here changing requirements are welcomed and very easy to change that's why developer can satisfy the clients. Use of prototype is huge. For better management , at the early risks are find out by developers. Besides there have some disadvantages those are very large number of documentation, there have lot of phases and management is not so easy.[4]. it is costly model for use. This model is mainly use for good risk analysis process that's why success is measured at risk analysis time ,when this analysis is done very perfectly that time project is successful otherwise project will not get success.[3]

Incremental Model: This idea was raised to remove the weak points of the water fall model. Here development is done with a small part of the project at a time and done with a cycle besides small part of the project day by day increment.[3]. First of all have to list all functional then find out the high risk functions. Then high risk function will be develop at early stages. Then have to give priority about customers needs and then after taking information from the customer then have to work on those functions first. The advantages of the process is that by using this process easily a developer team can analysis the risk and can easily reduce the risk by risk estimation. Besides there have some bad sides like when using this process that time project is not well organized and also planning is not so good. Also development process is long term. This process have to be use when there have risk count is very low that time it will be beneficial. This development process is costly and good for complex project. But good thing is requirements changing process is very easy and it is possible to change requirement later. It is easy to implementation and easy to maintenance. And the success rate of the process is high. [5].

Iterative Model: In this development process work is done part by part. Any development team can start their project by short collection of requirements and also can work with short part of the project. At the each iteration, development team present small version of the software. Until the final version release , this iteration will be continue.[4]. Here each iteration will release improve and more functionality of the software. Besides in each iteration part there have a testing phase. So overall we can say in each iteration part there have four common part those are 1.requirements collection from clients, 2.design and development of the software 3. testing of the software, 4.implementation. This iterative model is good for when all requirements are well known and very clear that time this process model is suitable. This process is not so good for all types of projects. There are some advantages of this development process like a client can get quickly feed back at this when using this model also clients can give a quick review. Besides we can say that this is client friendly because at any time client can change requirements and developer can easily accept the change and also requirements change cost is very low. An developer team can easily estimate the probable risks that's why very easy to estimation risk. Developer team easily measure the project progress. This system will be beneficial for a large project because in a large project there have many requirements that's why when developers are work for this project part by part that time it will be very efficient way and error less work.[4] . besides there have some disadvantage like you have to use huge resources for following this process. Besides there have no prototyping . this is not good for small projects.

Prototype Model: this is one of the light weighted developing process model. Here use iterative frame work. At each iterative phase testing done and also need users involvement after each iteration. The success rate of this model is medium but this is light weighted process. Besides in every phase overlapping happens. This process model is good for low to medium size project. Risk of this process is low and this model is flexible. Besides changing requirements is very easy besides documentation is not necessary but if anyone want then they can do. Timing of the

project is long term but here project team is small. Implementation of this process is very easy. Here release system of project is phase by phase[8]

The Rational Unified Process (RUP): this model is suitable for cross function project. There have no clash of any phase in this development process.by using this model , success rate is very high and heavy weighted process. Team size is large. Here use iterative and incremental process.at the developing time testing started. User involvement is needed in every iteration. This process is complex but very flexible.[8].

3. Agile Development Methodology:

In the software industry, there are several development processes here. The Agile Development process is one of them. It is rising now all over the software industry. The agile method apprehends as a lightweight and client-based method [17]. Several agile methods can be used combined in software development.

Agility is the authority to make or create and respond to change in order to gain profit in business conditions. Agile method is mainly an iterative and incremental development process [18].In Past iterative methods in every iteration length more than three months but agile method, every iteration length not more than 30 days. In different aspect, some exertion significantly up-down followed by incremental time boxed development, although others were more recomposed and iterative method simply for revisiting work but agile method not only just revisiting also evolutionary improvement and progression[18].There are so many names used for agile method such as lightweight lean methodology and so on and Barry Boehm the man who invent Plan-driven to distinguish to determine the spectrum from agile process[17]. Both processes are much better than other processes.

Agile and plan-driven development process can different spectrum of raising emphasis on plans, figure 1 clearly show us that the term of context documented process involve all the task like milestone plans, requirements collection, design[15]. Compared to unplanned hacking ,agile methodology do more amount of proper planning[15]. Fig 1 clearly shows that.

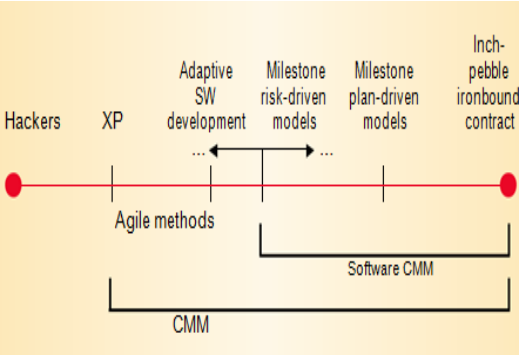

Figure 1.Planning spectrum[15].

Agile development process should be start with some fundamental components that
1.Development should be individual in each iteration and interactions with process and tools[17].
2.Execution software over extensive documentation[17].
3.Must be collaborate with customer in a interlocution[17].
4.Response to Change with actual or proper plan that following from [17].

Customer is the highest priority in agile development[17]. Highest priority is to satisfy all the client by delivery their valuable software early and continues.
Agile process give client freedom to change their requirement while development time will be lengthy, iteration time is to short that give client frequently update, business team and developer team should work together, support each other and give themselves motivation that project will be done as good as they work hard, simplicity, architecture, requirement, design emerge properly [17,10].

There are some principles that behind the agile manifesto.

Firstly to give main priority to the customers through early, easily and continuously giving feedback after each iteration. Then Client are welcome to change requirement even in late development phase[17]. This is the most advantage for customers. Every department of the project management must need to work together. Give a good vibes that make project environment motivate and trust them that project work will be done properly.
Agile methodology is mainly target to small to medium type software project[17].

The most challenging fact is collecting requirement. In 2001, 17 software developer introduce agile development process in based 12 principle on which various agile process such as extreme programming, Scrum and much more [10]. Although proper requirement collection is the main part to start agile methodologies. There are some prioritization techniques for requirement collection. Analytic hierarchy process that is generic decision method which prioritize requirement and work pair wise comparisons, prioritize cost and value on requirement perceptive,100 dolor testing or cumulative voting to collect requirement, use sorting algorithm is call planning game and a highly productive team that give achieving project success[10,16].

In this fast growing world, every country's digital economy depends on software domain area. And now agile methodology is the most preferable, planned, synchronize process that replaced other traditional development process, Extreme programming, scrum agile method and Feature Driven Development(FDD) process is the most popular agile process that produce most higher and quality level software in short period of time that is so much time consume and satisfy the client[19].

Scrum methodology is the most efficient and comprehensively best agile development process.
It's basically work in a three phases, one is pre development phase then development phase and lastly post development phase.
It's actually design for rapidly requirement changes request handle from client, it's running a project by collaborate with the team and communicate between each other and forwarding the project and break the whole project some sprint that is not more than 30 days.[19,12,13].
Here the fig 2 clearly show the life cycle of scrum development process.

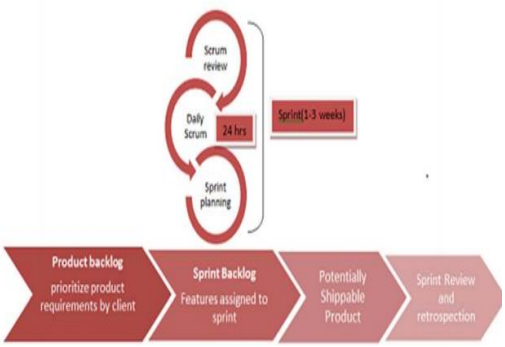

Figure2. Scrum life cycle[12].

In each sprint time noting can be changeable but after one increment done then client can add new requirement for the project [12]. Each sprint comprehend traditional phases of development, requirements and delivery phases. Every sprint must be planned with in one to four week, and development time must be in a specific time to complete a sprint. And scrum master called a scrum meeting daily to see progress of development. Term of quality, client demand, low budget and faster delivery, scrum methodology take all those privilege.

In software industry agile development process play an important key role player, it has so many benefits in term of requirement choice, Low budget project but quality product, customer requirement changes in the mine time of developing system and so on[13].
Its help developer to enhance their knowledge also help developer to find out wrong codes in the good time that they get in the mine time of each iteration[13].

Extreme Programming is one of agile software development process that also have some more benefit than other methodology that used in software industry. Extreme programming life cycle depends on five or six phase. One of is

exploring the software product that client discuss then planning to start or what will be the next iteration and collecting requirement from client. Thirdly Iteration plan that how to analysis the pair of programming that develop by developer then design a prototype for client. That will help client to visualize their project what will he get after finishing his project. And after releasing every iteration start to testing the product what will get. Release as a small iteration within 1 to 3 week. It will help client to know what will happening with his software product. Then maintain whole process and after every iteration start to planning, what will be the next step of next-iteration and release update. Finally end the project and handover to client. Also each iteration complete client can add or modify his added requirement or new one.

FDD mean Feature Driven Development which is one of short-iteration agile software development model process. This model was invented to do easily large scale projects.
FDD process work mainly in five process stage.
Process stage 1: Develop an Overall Model
Process stage 2: Build a Features List
Process stage 3: Plan By Feature
Process stage 4: Design By Feature
Process stage 5: Build By Feature

In process develop an overall model stage firstly defining project modeling, create object model as high level and must to create all the diagram. Then identify all the feature list and documentation of all the feature list in process build feature list. In this plan by feature process stage must build all the initial schedule for characterize business value of software and consider difficulty, risk. Mostly Time estimation and assign chief programmers[20]. After that formation of feature by feature teams and collaborate all the member of team to check all the design and analysis the level or quality. Investigation of code development process on design by feature process stage[20]. And lastly implement all the design feature on project. Testify all the feature level that bring project level high and investigate all the code implementation development[20]. Test all the code implementation by using unit test, integration testing and many more that need for project. And repeat all the testing process each of feature. Finally after all the feature processes are pass then project should be deliverable[20].
In fig 3 clearly show that the development life cycle about FDD process.

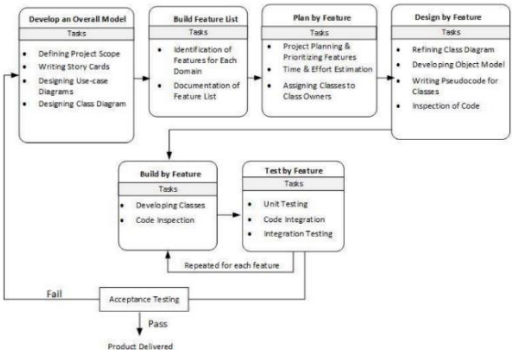

Figure 3. FDD development model cycle [20].

Agile methodology is light weight method that small software industry can use this methodology to more benefit then other traditional development process. Those development processes are takes more time than agile software development methods. Other traditional process like Waterfall development mode that is widely used methodology, which is heavy weight development process that takes more iteration time for 1st increment, there are several problem found in this process like change requirement that defects all others that makes development delivery late [14]. Most of the problems are occur when requirement collection and validation also verification are going and also this development process is used for large range development criteria that is make issues for waterfall model [14]. On other hand agile methodology is lightweight development process [17]. In term of frequent delivery agile model process takes short quality time, makes better quality product also productivity improvement and mainly focus client or customers satisfaction over their feedback about software and again customers have freedom to change their requirement that makes more benefits than other traditional methodology [17,9,11]. In fig 4 clearly shows that how much benefits of agile software development methodology have.

Figure 4. Agile development model benefits [9].

4. Agile Method's Issues:

Several issues were discovered through an in-depth case study for agile. An adaptive agile method in software development refers that agile method ensure the flexibility to adapt changes for a lot of time . The major purpose of agile solution delivery is to give value to an organization in supplement , which are accommodated to suit particular requirements and built over time. There is a biggest issue that is occurring in terms of developing software in an agile way is communication challenges [36].Communication problem is one of the root causes of project failure. Miscommunication occurs because of large team project. It is a source of bad communication. Because there are more members in a team, it is more difficult to engage with them all and share data and information. Factors such as poor communication among team members, a common knowledge of project objectives, and a lack of resources are all caused by a lack of commitment and teamwork. There are also issues with communication between development and the product owner, resulting in disorganized communication and a lack of documentation. The ability to achieve the goals of agile software development will be harmed as a result of poor communication organization. There are other challenges referred to as "day-to-day problems." For example, a large project with a high level of complexity, too many meetings, too many open issues, a lack of time to fix failing tests, and so on. Because the agile method for software development is so big, it is difficult to keep everything up to date. There will be some project complexity here as well. It is quite difficult to interact with each other because a major project has so much manpower in their team. There are too many meetings as a result of engagement, and there are too many open concerns. Lack of time to fix failing tests can occur when interaction is a problem. It could also lead to a lack of preparation[37]. The lack of management support is one of the main reasons why agile does not succeed in all circumstances. This happens because of departmental fragmentation, the lack of a single product owner's authority, and management's unwillingness to change their behavior, among other things. In that field, experience counts for a lot. Giving up too soon can be a problem if you don't have enough experience making it work. There are also product owners who are inexperienced. The ability of the development team to make independent judgments is a challenge in agile software development [38].The system has testing lead times and maintenance issues. Testing lead times are extended as products that should be delivered might not loose the quality or even the project is late. During the case study, additional concerns that had not been specifically identified in the literature appeared. Due to previous working of a company there are a lot of documentation remained. Some of them are trash as there is no use of them. The ambition is to reduce them as they are unnecessary because they are quickly outdated while other documents can be replaced. Sometimes important documents are also be removed carelessly that results in decreased visibility of project progress. As there are lot of team members, sometimes decisions are came like there are too many unplanned tasks. Then there might be a big problem arise the software development process. New to agile is one of the common reasons there.Technical issues that can be related to or producing ordinary processes without being subjected to special mechanism. Automation by which the system operate and automatically controlled by technical devices and general testing. Agile provides numerous processes and tools for managers to use in bad faith.Communication with customer and client and can easily expressed the whole issue. Generally agile development works well when the team is at one place works together clinically that is all team members carry out the required practices at one site only. Agile testing unit can be applied to software but embedded systems come into play then it becomes a difficult task. In agile process, continuous management of requirements is a challenge since not all of them are fixed.As there are a lot of team members and lot of meetings can occur, the management process tend to be vulnerable. So it cannot work as per as clients requirement. Therefore the satisfaction level of the client is not perfect. As this method has this kind of issue, people are now trying to compare it with other software development methodologies

[39].Although agile software development methodology includes a set of procedures and quality standards that are ideal for software deployment, it can be difficult to recognize all of the drawbacks that it has. It is critical to have the backing of any company's management team to assist alleviate any problems or concerns that may arise.

5. Success Factors of agile:

success element is delivered as an method which detects names and compare software performance [26]. on this segment, point out nearly all of the fulfillment factors. Agile software program improvement technique (ASD) became originated by means of 17 software practitioners in 2001. in keeping with the literature, consumer plays a totally critical role in the achievement of a fast-paced software assignment. consumer involvement inside the venture may be categorize as client collaboration, patron delight and purchaser commitment.

one of the quickest methods to get a client first is to supply critical software program in advance of time and on an ongoing foundation. according to a look at performed with the aid of purchaser pleasure followed by means of customer collaboration are taken into consideration to be the maximum important elements in determining a undertaking achievement. As ASD strategies evolve and replicate, the patron can be an crucial role participant in the course of the lifestyles cycle of the project. The purchaser plays a responsible function inside the mission through validating the software and prioritizing the requirements for every duplication. So, customer dedication is likewise an critical factor in success [25].

The Agile manifesto emphasizes the value of the individual and the interaction of processes and tools. A very fast project requires a very efficient environment to be successful. Agile projects rely heavily on customer engagement and continuous feedback. The organization must therefore be strong and flexible in order to change. Close cooperation between team members is also needed [25].

Typical software methods rely on written systems but the fastest methods prefer internal planning and control within a team rather than interrupted by external managers. The level of independence in planning and management is therefore an important factor in success [25]. The success of an ASD project is often linked to human resources. Human resource factors are also considered to be key factors in the success of an ASD project. Some of the aspects of the ASD project success that will be addressed below can be equally categorized as human characteristics. Competence means whether a person has real world knowledge in the field of technology, has developed similar programs in the past, and has good communication and interpersonal skills. Knowing also means that a person has acquired the necessary skills and abilities to do a good job overtime in the same systems. By using the skills demand matrix as a skills model, organizations can access any skills gaps in ASDT members and categorize members who will need additional training in a particular skill [28].

Most researchers agree that People Dimension plays a key role in any software development project. People ratings are divided into eight components, and these eight factors are shown in Figure 1. Human characteristics may include Education where the team should learn the fastest strategies and how to use them and adopt them in slow companies, achieved by learning to support teams, and participate in all aspects. Also, team member and supervisor should be committed to their tasks and project [27].

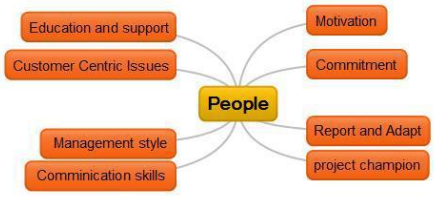

Figure 5. Identify and Classify Critical Success Factor of Agile Software Development Methodology [27].

As an organization begins to evolve, the organizational culture and ideology must change. Speed is a process that brings new practices to teams and managers, and often quickly affects the culture and mind of the organization. Although these factors are important, changing everything at once can be a great challenge for an organization. The size of the organization and its properties is shown in Image-2 [27].

Figure 5. Identify and Classify Critical Success Factor of Agile Software Development Methodology [27].

The success of an Information Technology (IT) project depends on having a project manager with effective decision-making, leadership, and project management skills (PM). The success of the project also depends on completing the project with a given budget, time, and scope. But, there is a limited understanding of Agile leadership Effectiveness in main excessive overall performance team [31].

Product management software development is a major problem for software firms, where large drivers are low cost and short time to go to market. To manage productivity successfully, it is important to identify the most appropriate difficulties and develop coping strategies. Agile software (ASD) software development methods, such as Extreme Programming and Scrum [31], have emerged as ways to reduce the complexity of the software development process, which may lead to product development. They focus on reducing development time or cycle and manage the inevitable shifts caused by market forces [31].

The success of an ASD project is often linked to human resources. Human resource factors are also considered to be key factors in the success of an ASD project. Some aspects of the success of the ASD project that will be considered below can be equally classified as individual traits. In addition, Cao research [31] identified aspects of personal success that include:

(a) Team members with high potential and expertise.
(b) Members of the party are highly motivated.
(c) Managers who know the fast-track system.
(d) Managers exposed to light or flexibility.
(e) Management style; (f) Integrated, and co-operative team.
(g) good customer relations. However, in preference to moving into the trivial matter of the class, the researcher will honestly talk to them [31].

The organization has a positive impact on the success of this project. Culture can shape many things in an ASD project. Various researchers have shown that organizational factors are also important factors in success for a successful project. These issues will also be discussed including customer commitment, decision-making time, team distribution, business culture, planning and control, and most important business / security priorities.

Cao research found that a successful organization has the following characteristics[31]:

(a) Strong support of officials.
(b) A committed sponsor or manager.
(c) The culture of co-operative rather than governance.
(d) The negotiated culture sets a high value for face to face.
(e) Communication.
(f) Organizations where the approach is universally accepted.
(g) Consolidation of the whole group.
(h) An institution with an appropriate speed-style operation.
(i) An accelerated reward system.

This paper is important for both researchers and physicians because it can illuminate SFs as an important factor in the development and control of high-performance ASDTs[31]. It also commonly contributes to the savvy development of SFs in particular and non-conventional undertaking control modes, which might be utilized in ASDTs [25]. Research is important for project managers and national project coordinators and their project teams because their findings, when integrated into training programs, may lead to better understanding and application of SFs. If monitoring of ASDTs projects actually improves project success, there is a lack of information about SFs and their links to project success.

These findings provide an important step in studying how success factors are important in building and managing high-performance ASDTs. Organizational managers and staff from the various industries that have ASDT management can benefit from our research findings through or by developing strategies or processes that can improve the performance of ASDT for sustainable long-term development. ASDT High Performance Outcomes

include fully enhanced team performance, market time, project success rate, and safe and healthy organizations or communities.

6. Methodology:

A mixed-methods strategy was used to accomplish the study's goals. A broad assessment of academic and commercial publications on agile software development projects was part of the research's initial phase. The objective of the review was to list and compile the crucial elements that make agile software development projects successful. Case studies of productive agile software development projects were examined in the second stage of the study. The case studies were used to obtain a deeper knowledge of how the important success variables were discovered and interacted with in actual projects. The case studies offered an opportunity to investigate how businesses might successfully apply agile approaches and raise the success rate of their projects. The case studies were carried out utilizing a qualitative research methodology, with semi-structured interviews serving as the main data collection technique.
Project managers, developers, scrum masters, and other important stakeholders were among those who were chosen as participants based on their experience with successful agile software development projects.

Data Analysis:

In every phase of agile development methodology testing happen continuously and problem arrives. There are so many types of problems that occur in the development phase also testing phase. And most of problems come out at the time of the testing also development process. In poor resources for requirement and planning is one of the major problem that arrives in the agile development method. More or less there are mostly five types of problems or disadvantages that can occur like so much less predictability about requirements, more time but commitment, client or customers have greater demand, limitation of documentation, and lastly falling off track. In the process of testing some issues come out that talk about lack of information and limitation on documentation. In the phase of testing, there are so many testing methods that can be applied to the agile development process. Such as unit testing, integration testing, regression testing, black-box testing, white box testing, alpha testing and many more go on. In every testing begin there are so many problems or bugs that come out and developers try to solve those problems and also try to finish their work properly. And so many requirement change requests from clients but lack of time is one of the main problems in agile development. And also instead of waiting for the development phase to finish, the testing process has been started. In every sprint of agile development finish, clients are added new features or change their requirements so the testing process continuously happens. So testing is so much important for further improvement of software projects cause if testing fails then so many bugs will occur and development can not be possible properly.
Respond 3 ‘Low predictability, unpredictable requirement’
Respond 5 ‘change management
Respond 3 ‘Test story, test plan, test case, test environment, test execution, bug report’
Respond 2 ‘instead of waiting for finish development testing process start and continuously arrive features are added’
Agile methodology actually enhance and help to continuous iteration of development and testing for any kind of agile software development project. The agile development process is a high acceptance of feedback and change requirements. In this method, clients are allowed to change their requirements after every iteration is complete. In the process, customers are the main priority over other SDLC processes. But on the other hand waterfall model is the old traditional software development process. In this process whole project is broken down into linear sequential phases where everything depends on the previous delivery of development. On other hand, agile doesn’t need this. There are so many problems that occur when the development phase starts like bugs only can fixable in during the phase but on agile after every sprint finish client can change their requirement also if any types of bugs occur they can fix those problem. In the agile development process testing and development, phases are more concurrent than our traditional waterfall model and other SDLC also. In the agile development process communication or connection between client and developer are more accomplished than in any other SDLC process. The waterfall model is not flexible and desirable for complex software projects where clients can’t be allowed to change requirements as much as they want or frequently. And most importantly testing period has been done lately for that reason bugs and problems occur in the phase of development and after development. And clients are not appreciable of change requirements or adding new requirements. But in the agile development process, there are huge acceptance of change requests and so much flexibility, and this flexibility is the proper reason that developers are using the widely agile development model process.

Respond 4 ‘due to change request’
Respond 2 ‘this methodology grants more communication or connection between customers/client, developments, managers and testers’
Respond 3 ‘Cause of flexibility’
Respond 6 ‘For customer engagement’
There are so many types of agile development that models are used in the software industry. Such as Scrum, Kanban, extreme programming (XP), Crystal, Dynamic Systems Development Method (DSDM), and many more. Although there are so many variants of agile methods are here but scrum and XP model is mainly used widely in the software industry. The reason behind using the scrum model is so simple that is customer communication with the developer. And mostly this process is based on iteration and the incremental way that client can change the request of requirement. Scrum development process is so much flexible that client can get so much value throughout the development process for that reason every client get satisfied. Throughout the development phase, client can add new features after completing each sprint. So the flexibility of scrum is the main reason for developers to choose the scrum development model. And extreme programming is kind of a lightweight development methodology that involves facilitates with good planning and small teams of the developer can achieve a high-ended quality product that helps to impress clients also customers, and also can enhance rapidly changing requirement responses that can help to satisfy clients.
Respond 3: ‘Scrum method’
Respond 4: ‘Scrum, XP’
Software development life cycle termed SDLC is a method of developing high-quality software. It is a software development process. It has some important phases like requirement analysis, maintenance, design, and testing that we can develop software systemically. It acts as a mentor to the project. It provides a better framework. The system can make sure timely and correct delivery of software to the client. It has a phase termed feasibility study. It examines the feasibility of the system. During the project life cycle, a software requirement specification document is made. This document includes everything which should be developed during the project. The system can make sure economic feasibility that the project can becompleted within the budget and it also fulfills customers’ expectations. As it is the process of planning, testing, and deploying, it is easy to get ahead of taking on a large project and very easy to track a project.it also helps to deliver the product to customers’ requirements without any bugs. A loss of a project member can reduce the development of a system. Hence the SDLC is a well-designed system by which everyone will be in order so that a new member of theproject can continue the process without any complication. People who put theeffort into understanding SDLC would a worthy investment. For more business and desired results, planning and analysis are a must. SDLC can assure a better perspective of system analysis and planning. SDLC can reduce the cost of software development. It also improves the quality simultaneously and decreases production time. Thus SDLC is a significant process in the software industry.
Respond 5: ‘To develop a systematically way’
Respond 6: ‘It is a process used by the software industry to design, develop, testing high.’
The agile method is one of popular methods for software developmen . This method can assure customers satisfaction. This method is more relevant for 'clients perspectives as working software indicates the project progress better than documents presented to clientlive meetings . It can also manage customer collaboration to provide more detailed requirements over contract negotiation. As the development process is running and changeable throughout the cycle, new features can be added without the need to start the work from starting. The customer has space for changes requirements that’s why clients are happy to get through the process. An agile based project is easier to understand for non-technical people. Besides clients can easily explain and share the overall development process and vision of the project with the developers. Also, they can share the relevant details and ideas that how can improve. This method has free space for features implementation. As a result, the software gets improved until it fulfills the customer's satisfaction and expectations. Activecustomer s can engage with the developers about product requirement discussions. As it is an iterative process so that there is enough scope to make changes tothe project at any stage. Although the agile process is a customer friendly process. Customers are always involved in this process by making decisions. It has better flexibility which can be so manageable. The improvement in this system is continuous and previous mistakes will not be repeated. It is so user-friendly. As all the facilities agile brings, it has better popularity. More and more companies are opting for agile worldwide because it is more relevant to customers. Also because of some non-technical clients, it has got better customer satisfaction with the product or the client perspective. That's why the method is so popular.
Respond 3: ‘due to flexible and easy to change requirements’
Respond 6: ‘Mainly for nontechnical clients’

Applying agile development process on a complex and large software project, it depends on what type of project it is. In most of the complex and large software projects have so many terms and many types of conditions that create much more complexity than other simple or small or medium type software projects. But yes, there is nothing impossible here, sometime agile method is much more reliable than other SDLC to complete the project. But the agile model helps when it comes to development cause after every iteration is a complete developer and client can discuss the development of a project which makes the project much more reliable and also create a good bond between client and developer. In the agile development process model, there is proper scalability that performs better than other SDLC methods. And depending on the project may agile development method is so much well defined for developing software projects. For this reason developers prefer agile development in a software project. In that case agile development process model is also preferable for large and complex projects.
Respond 3: 'It's depend on what type of project'
Respond 6: 'Depend'
Respond 1: 'Yes'
Agile development technique promotes a flexible approach to product planning and implementation so that organizations can quickly respond to feedback. Agile Methods provides software project managers with an alternative development and management solution that provides excellent support for projects with unspecified or rapidly changing needs. At first, agile development process provides a nice road map for product management. By following this road map developers can progress their work rapidly and also they can show their improvement of work. Agile method gives customers the product they want, for every stage they give the customer the right to check their requirement, and if want can change it. At the final stage they can do it, so they find their product as they aspect. Product managers have a responsibility to create a product strategy in partnership with stakeholders and to protect the purchasing and coordination of the organization. These product elements are aimed at pleasing customers, displaying product value, building trust, and generating revenue.
Respond 2: 'Customer satisfaction
Respond 3: 'Product acceptance'

Discussion:

Data analysis about SDLC state that software development life cycle is the method for develop high quality software product by a systematic way. It has so much phases such as requirement analysis, documentation, design, testing, development on systematic way. On other hand this is one kind of process or method that the clients are well understood about their software project and how it will be increase also develop and the overall development process go on[1]. Agile development process methodology is one of the best software development process. In software industry there are so many variant of agile method used to develop software project. But the analysis state that majority of software developer suggest or prefer to use scrum methodology to develop software project. The main reason to prefer scrum development method that is customer communication with developer and change request. But also so many developer address that Extreme programming and Feature driven development are also produce most higher and quality software product[19,20]. Scrum development process is so much flexible that client can get so much value throughout the development process for that reason every client get satisfied. Though XP based on high programming but also work based on six phases that involves facilitates with good planning and small teams of the developer can achieve a high-ended quality product. In the FDD technique, the first step is to define project modeling, then develop a high-level object model and all of the diagrams[19,20]. Then, in the process of building a feature list, identify all of the features and their documentation. In this plan by feature process stage, all of the first schedules for determining the business value of software and envisage difficulty and risk. The majority of the time, time estimation and assigning chief programmers[20]. In every SDLC there is so many testing method are using for good quality software product. But in the testing phase there are so many problem occur or come out also development phase. Data analysis state low predictability, unpredictable requirement also continuous change request also change management liable for arrive so many problem in agile development process method. During the testing procedure, various concerns arose that discussed a lack of information and documentation limitations. In the phase of testing in agile development method, there are various types of testing methodologies can be used. Unit testing, integration testing, regression testing, black-box testing, white-box testing, alpha testing, and other types of testing exist. On the other-hand paper or literature state that inadequate or insufficient experience with agile development methodology, there was a lack of knowledge of broader organizational change that was required also when a company's philosophy or culture clashes with agile values, it's a deftness for disaster[35]. Also low stability of requirement responsible for failing or occur problem in agile development. But in the term of customer satisfaction agile development method is the most powerful SDLC. The more the client can be involved in the day-to-day software project operations, the better the communication will be and the requirement for intermediate papers or

documentation will be reduced[35]. In analysis state in this agile process, client or customers are the main priority. On the other hand old traditional process like waterfall model that breakdown whole project into linear sub-sequential phases where every single thing depend on previous development lor deliver[2]. In agile due to flexible and easy to change request, customers also satisfy also they can change their requirement in middle of project development when every iteration finish. Agile methodology is mainly based on iteration and incremental development method. In this present time many software company used agile on their large and complex project also. Applying agile method on complex project depends on what type of project it is [26,27]. Analysis state depending on software project, proper scalability and well sufficient experience with agile method can be appliable for complex and large project. The agile development process is a flexible approach to product management and deployment that allows companies to respond quickly to customer feedback. Also these product features are intended to please customers, demonstrate product value, foster confidence, and generate income. Organizational aspects have also been proved to be major elements in project performance by several researches. Customer commitment, decision-making time, team distribution, corporate culture, planning and control, and the top business and security concerns will all be reviewed[31].

7. Conclusion: In conclusion, the research on multidimensional success factors of agile software development projects has identified key elements that contribute to the success of such projects. Effective communication and collaboration among team members and stakeholders, flexibility to adapt to changing requirements, active involvement of customers, technical competence, and team composition are all crucial success factors. The study highlights the importance of managing people, processes, and technology in the successful implementation of agile methodologies. Clear communication, collaboration, and trust within the development team are essential for delivering projects on time and to the expected quality. The involvement of stakeholders, especially end-users, improves the alignment of project goals with the business objectives. The use of appropriate tools and techniques such as continuous integration and continuous delivery can enhance the efficiency and effectiveness of the agile development process.

These insights can be applied by organizations, development teams, and project managers to improve the outcome of their agile projects and contribute to the ongoing evolution of the agile development methodology. This thesis provides a comprehensive understanding of the multidimensional success factor of agile software development projects and the findings can be used by organizations to improve their processes and by future researchers to expand upon this work.

In summary, this thesis has emphasized the need for organizations and development teams to focus on multidimensional success factors in agile software development projects in order to deliver successful outcomes. The research provides valuable insights into the key elements that drive success in agile projects and can be utilized to improve the outcome of future agile projects.

Acknowledgment: The authors of this thesis, Mahiul Alam Sad, Shionty Ghosh, Punam Ghosh, Fahim Faisal, would like to extend their heartfelt thanks to all those who have supported them in the completion of this project. In particular, we would like to express our gratitude to their supervisor, Dr. Dip Nondi, for providing constant guidance, encouragement and invaluable insights throughout the entire project. Finally, we would like to acknowledge American International University-Bangladesh(AIUB) for providing the resources and opportunities necessary to pursue this project. It has been a privilege to work with such a talented group of individuals and a great pleasure to have the opportunity to complete this thesis.

REFERENCE:

[1] Scroggins, R., 2021. SDLC and Development Methodologies. [online] Computerresearch.org. Available at: <http://computerresearch.org/index.php/computer/article/view/148>

[2]balaji, s., & Murugaiyan, D. M. S. (2012). WATEERFALLVs V-MODEL Vs AGILE: A COMPARATIVE STUDY ON SDLC,vol. 2, 1–5.
[3]T Bhuvaneswari1, & S Prabaharan2. (2013). A Survey on Software Development Life Cycle Models, Vol 2, issue.(5), 262–267.
[4] S, Shylesh, A Study of Software Development Life Cycle Process Models (June 10, 2017). Available at SSRN: https://ssrn.com/abstract=2988291 or http://dx.doi.org/10.2139/ssrn.2988291
[5] Alshamrani1, A. and Bahattab2, A., 2015. A Comparison Between Three SDLC Models Waterfall Model, Spiral Model, and Incremental/Iterative Model. IJCSI International Journal of Computer Science Issues, Vol. 12(Issues. 1).
[6] Rahim, M. S., Chowdhury, A. Z. M. E., & Das, S. (2017). Rize: A proposed requirements prioritization technique for agile development. 2017 IEEE Region 10 Humanitarian Technology Conference (R10- HTC).
[7] Mishra1, A. and Dubey2, D., 2013. A Comparative Study of Different Software Development Life Cycle Models in Different Scenarios. International Journal of Advance Research in Computer Science and Management Studies, Vol. 1(Issues 5), pp.1-7.
[8] Amlani, R. D. (2013). COMPARISION OF DIFFERENT SDLC MODELS. International Journal of Computer Applications & Information Technology, Vol. II(Issue I), 1–5.
[9] Matharu, G. S., Mishra, A., Singh, H., & Upadhyay, P. (2015). Empirical Study of Agile Software Development Methodologies. ACM SIGSOFT Software Engineering Notes, 40(1), 1–6.
[10] Rahim, M. S., Chowdhury, A. Z. M. E., & Das, S. (2017). Rize: A proposed requirements prioritization technique for agile development. 2017 IEEE Region 10 Humanitarian Technology Conference (R10- HTC).
[11] Syeed, M., Hayat, R. and Miah, J., 2021. Agile Fitness of Software Companies in Bangladesh: An Empirical Investigation. International Journal of Advanced Computer Science and Applications, 12(2).
[12] Srivastava, A., Bhardwaj, S., & Saraswat, S. (2017). SCRUM model for agile methodology. 2017 International Conference on Computing, Communication and Automation (ICCCA).
[13] Altameem, E. (2015). Impact of Agile Methodology on Software Development. Computer and Information Science, 8(2).
[14] Petersen, K., Wohlin, C., & Baca, D. (2009). The Waterfall Model in Large-Scale Development. Product-Focused Software Process Improvement, 386–400.
[15] Boehm, B. (2002). Get ready for agile methods, with care. Computer, 35(1), 64– 69.
[16] Fatema, I., & Sakib, K. (2017). Factors Influencing Productivity of Agile Software Development Teamwork: A Qualitative System Dynamics Approach. 2017 24th Asia-Pacific Software Engineering Conference (APSEC).
[17] Paulk, Mark C (2018): Agile Methodologies and Process Discipline. Carnegie Mellon University. Journal contribution.
[18] Larman, C., & Basili, V. R. (2003). Iterative and incremental developments. a brief history. Computer, 36(6), 47–56.
[19] Livermore, J. (2007). Factors that impact implementing an agile software development methodology. Proceedings 2007 IEEE SoutheastCon.
[20] Nawaz, Zahid & Aftab, Shabib & Anwer, Faiza. (2017). Simplified FDD Process Model. International Journal of Modern Education and Computer Science. 9. 53-59. 10.5815/ijmecs.2017.09.06.
[21] Jeffrey A livermore," factofrs that impact implementing an agile software development methodology" IEEE 2007
[22] Singh,A.2012. Agile: Analysis of its problems and their solutions.International Conference on Recent Advances and Future Trends in Information Technology.
[23] Vasudeva Shrivastava, S. and Rathod, U.,2014.Risks in distributed agile development: A review. ELSEVIER
[24] Tomanek,M. and Juricek, J.,2012.ADDRESSING MAIN PROBLEMS IN PROJECT MANAGEMENT(PRINCE2) BY ADOPTING AGILE PRINCIPLES(SCRUMS). IADIS International Conference Applied Computing.
[25] K.Beck, Extreme programming explained: embrace change. Addison Wesley Professional, 2000.
[26] D.J. Reifer, F. Mauer, and H. Erdogmus, "Scaling agile methods," software, IEEE, vol. 20, no. 4, pp. 12-14, 2003.
[27] Ahmed Awad, Mohamed. A Comparison between Agile and Traditional Software Development Methodologies. U of Western Australia
[28] Ullah, Mehar. Comparison and Problems between Traditional and Agile Software Development Methods. Lappeenranta U of Technology, 2014
[29] Webpreneurs Hub. (2018, April 20). Retrieved from Latest Technology Trends in Nepal:http://webpreneurshub.com/latest-technology-trends-in-nepal/.

[30] Anand, J. V. "Design and Development of Secure and Sustainable Software Defined Networks." Journal of Ubiquitous Computing and Communication Technologies (UCCT) 1, no. 02 (2019): 110-120.
[31] T. Dyba and T. Dinsoyr, "What do we know about Agile software development," IEEE Software, pp. 6-9, 2009.
[32] M. Fowler, "Using an Agile Software Process with Offshore Development," 2006. [Online]. Available: http://www.martinfowler.com/articles/agileOffshore.html
[33] Langer, A. M., "Guide to software development designing and managing the life cycle". London ; New York, Springer,: 1 online resource (xvii, 350 p.), 2012.
[34] Martin, R. C., "Agile software development : principles, patterns, and practices",Upper Saddle River, N.J., Pearson Education, 2003.
[35] Miller, Gloria. (2013). Agile problems, challenges, & failures.
[36] Boehm, B., & Turner, R. (2003). Observations on Balancing Discipline and Agility. Agile Development Conference, (pp. 32 - 39). Los Angeles, CA.
[37] Tomanek,M. and Juricek, J.,2012.ADDRESSING MAIN PROBLEMS IN PROJECT MANAGEMENT(PRINCE2) BY ADOPTING AGILE PRINCIPLES(SCRUMS). IADIS International Conference Applied Computing.
[38] Beck, K., Beedle, M., van Bennekum, A., Cockburn, A., Cunningham, W., Fowler, M., Grenning, J., Highsmith, J., Hunt, A., Jeffries, R., Kern, J., Marick, B., Martin, R., Mellor, S., Schwaber, K., Sutherland, J., Thomas, D.: Manifesto for Agile Software Development
[39] K. Mannaro, M. Melis, and M. Marchesi. Empirical analysis on the satisfaction of it employees comparing xp practices with other software development methodologies. In Proc. of the 5th International Conference on Extreme Programming and Agile Processes in Software Engineering (XP 2005), pages 166–174, 2004.