Study on Agile Story Point Estimation Techniques and Challenges

Ravi Kiran Mallidi

School of Computer Applications
Lovely Professional University
Punjab, India

Manmohan Sharma

School of Computer Applications
Lovely Professional University
Punjab, India

ABSTRACT

Agile Scrum Estimations are incredibly crucial for planning and managing Agile Scrum teams. Most of the Agile projects overrun and budgets due to a lack of proper estimates. Lack of proper estimates leads to imprecise planning, design, development, and management of the project, which may deliver the unsatisfied product/project to the customer and thereby unsatisfied customer/end-user. The project risks were not identified when the initial phase of the project leads to indecorous estimates. The primary objective is to present various Agile Story Point estimation techniques used in development environments and their challenges. Helps the development teams to adopt proper estimation methodology in their projects. Various estimation techniques are available to estimate the Agile projects. Various Agile estimation models are Planning Poker, T-Shirt Size, Dot Voting, Bucket System, Large / Uncertain / Small, Ordering Method, Divide Until Maximum Size, or Less. The paper elaborates on each of the Scrum estimation models and includes challenges for each Agile estimation technique.

Keywords
Agile, Scrum, Estimation, Story Point, Planning Poker, T-Shirt, Dot Voting, Bucket System, Dot Voting.

1. INTRODUCTION
Agile Software development advocates adaptive planning, evolutionary development, early delivery, continual improvement, and flexible change in requirement at any time of the development process [25]. In 2011, the Agile Alliance created the *Guide to Agile Practices,* an open-source compendium. The guide defines working definitions of agile practices, terms, and elements, along with interpretations and experience guidelines from various communities worldwide. The Manifesto of Agile Development comprises of 12 principles. The principles are Continues Delivery, Adopt Requirement Change, Continues Delivery, Cooperation between Developers and Business daily, Individuals Motivation towards work, Face-to-Face Conversations/Meetings, Working Software, Development Sustainable teams, Continues Design, Art of Maximizing the work not done, Best Architectures, and More effective teams.

Agile estimation is done at different levels of the project life cycle. Estimations are performed at Proposal Level, Release Level, and Sprint Level. Sprint Level estimations are very critical for Scrum Master to deliver the product on time with maximum quality. Management teams track the efforts to properly plan and managing the teams in terms of project implementation (development), testing, and delivering the product/project to the customer/end-user within the specified timeline. If the estimates are not accurate, the managers/scrum master in the Agile project leads to compromise on planning and management activities, which leads to bad quality or non-desired product/project to customer/end-user. Agile has different software development methods to support a broad range of Software Development Life Cycle (SDLC), which includes:

- ✓ Agile Unified Process (AUP)
- ✓ Extreme Programming (XP)
- ✓ Adaptive Software Development (ASD)
- ✓ Feature-Driven Development (FDD)
- ✓ Kanban
- ✓ Scrum
- ✓ Scaled Agile Framework (SAFe)

Agile has different software development practices to support requirements, design, coding, testing, planning, risk management, process, and quality. Below are some notable Agile Software Development Practices:

- ✓ Acceptance Test-Driven Development (ATDD)
- ✓ Backlogs
- ✓ Behavior-Driven Development (BDD)
- ✓ Pair Programming
- ✓ Continues Integration (CI)
- ✓ Incremental Development (ID)
- ✓ Test-Driven Development (TDD)
- ✓ User Story
- ✓ Planning Poker
- ✓ Velocity Tracking

Scrum is one of the prominent frameworks in Agile projects for the past few years. Scrum uses for developing, delivering, and sustaining complex products. Sprint is planning in the Scrum framework to break the work into goals that can complete within the time frame of 2 to 3 weeks [22]. Software Development term *Scrum* [26] used in the Agile framework for the Iterative and Incremental process to managing intricate work.

As part of the Scrum process, the scrum master conducts the Spring Planning session, and the development team pulls out the tasks as per priority and estimates using different methods adopted by the team. The size of the Product Backlog Item (PBI) is defined using Story Point. Teams across the Organizations use various Agile estimation techniques used in their project scope and their experience. The Scrum Master job is to adopt the most suitable estimation model to estimate the tasks. Scrum helps the team to estimate the Story points more accurately. Relative sizing is one of the significant aspects to do Story Points estimations by the team members. All team members, Product Owner, Scrum Master, Scrum Developer, Scrum Tester, and Stake Holders are responsible for estimating the efforts.

This study compares different Story Points estimation models Planning Poker, T-Shirt Size, Dot Voting, Bucket System, Large / Uncertain / Small, Affinity Mapping, Ordering Method, Divide Until Maximum Size or Less elaborated along with their challenges.

This paper is organized as follows: Section 2 describes Literature Review, Section 3 describes different Story Points Estimation Models with challenges, and Section 4 concludes the paper.

2. LITERATURE REVIEW

There are studies about effort estimations modes in literature. Some of the studies in the literature are briefly summarized below.

Agile is using by the organizations for the past two decades, and different trends and evolution are in place. Agile uses the Scrum framework is one of the best. Agile development and maintenance project challenges are Iterative development, focused work objective, teamwork, customer involvement, face-to-face communication, light documentation, frequent testing, collective ownership, and knowledge transfer [13]. They compared the Traditional and Agile estimation models. Discussed Non-Algorithm estimation methods like Expert Judgement, Analogy technique Wideband Delphi, and Algorithm based estimation methods like Source Line of Code (SLOC), Functional Point, Object Point, Constructive Cost Model (COCOMO) [19]. COSMIC (Common Software Measurement International Consortium) Functional Point Measurement (FMS) methods used to estimating mobile development estimates. Developed View Functionality, Data Manipulation Function, Inquiry Function, User Support Function, and Specific Function types Functional process measurement for Agile estimations [16]. Compared various estimation models like Use Case Point, Functional Point, COCOMO, Algorithm-Based Model, Expert Judgement Model, and Estimation [20].

Compared with the Traditional and Agile models in terms of Adaptability and Teamwork. Risk of Agile measured in terms of Lack of Understanding Requirement, Team Meeting, Visit, Training, Poor Communication, and Number of project Personnel [2]. A proposed hybrid approach to achieve better realization of low cost with compared the characteristics of development methods includes Large Scale, High reliability, High productivity, High estimation accuracy, Early realization, and Easy of change [25]. Five multiagent core components estimation systems were developed. Ontology knowledge-base, knowledge creation, knowledge tree maintenance, and facility factors significantly improve the effort estimations in Agile development [1]. Accuracy of effort estimates improved by using expert-based estimations in Agile projects [4]. Constructed reference model for estimations by examining the story point life cycle and combined the most commonly used story points [5]. Studied different SDLC processes from Waterfall to Agile and provided advantages and disadvantages of each of the approaches [12]. The Algorithm proposed to estimate by considering the environment and technical factors to achieve accuracy. The Algorithm consists of Total Technical Factors (TTF), Total number of Use Case points (TUCP), Total Environmental Factors (TEF) weights for calculating the Total Estimated Cost (TEC) [17]. Calculate Functional Point for a given requirement with Weighted and Complexity Factors. Add the infusing factors for the FP estimates. Proposed Algorithm to convert Functional Point Estimates to Story Point Estimates [29]. Agile projects success rate is more than Waterfall (around 2X more), 42% Successful, 50% Challenged, and 8% failed for Agile projects. Agile projects success rate is 60% greater than non-agile projects from The Standish Group 2018 Chaos Report [27]. Discussed various estimation methods like Analogy Based Effort Estimation, Regression-Based Estimation Model, Software Sized based Estimation Model, Functional Estimation Model, Work Breakdown Structure, Story Point Estimation and suggested WBS with Story Point was the best fit for Agile projects [24]. Comparing the average of suggest User Stores and Consensus of the size of User Stories and concluded using average size causes less accuracy while compared to Consensus [9]. Surveyed Agile mobile application development and observed that 63% of projects are using Poker, 47% use Analogy, and 38% use expert judgment [3].

3. AGILE SCRUM ESTIMATION MODELS

Agile manifesto uses various types of Agile Scrum estimation models. Agile projects often challenged/failed due to aligning project constraints like Specific Scope, Time-bound, Estimated Budget, Quality Constraints, resource attitudes, and Projects Risks to estimate the product accurately. Story Points are estimated by using comparative/relative sizing. Analyze all the tasks/stories, and identifying the reference task/story is base for relative sizing of estimation of Stories. For this, pick up the backlog tasks and brainstorm within the team and ask questions with the Product Owner and understand the requirement clearly. List the activities performed for completed the task/user story by proper Design, Code, Unit Testing, Integration Testing, and Acceptance Testing. Make the relative sizing Story Points of the selected task by consensuses with all stakeholders. Select another task from the backlog and assign the same amount of work as assigned to the recent Story if the Story requires the same amount of effort to complete. Task/Story requires more effort, assigns the higher values, and is the Task/Story requires less effort to assign the lower value based on the relative Story Point. Below are the widely used estimation models for Scrum Story Points derivation for better estimates. Based on the Story Points, the team delivers the specific Stories for each sprint cycle to achieve the working model. At the end of each sprint cycle, the working model is shown to the end-users for feedback. The feedback from the end-users is incorporate in the consequent Sprint as a new requirement.

3.1 Planning Poker

Planning Poker is a widely used estimation techniques in Agile Story Points estimations. JIRA is having one of the most prominent tools for managing Agile Poker projects. Planning Poker is an agile estimation technique that establishes relative sizing using story points and playing cards with notations of 0,1,2,3,5,8,13,21,34, and 55. These playing card values represent story points for a particular item/story measured by the team in the scrum meeting. The number sequence usually the Fibonacci sequence. The product owner or scrum master reads the user story and explains all the features and requirements and discussions among the team for various technical and non-technical elements for estimations. The product owner primary responsibility is to clear all the questions or clarifications asked by the team. After detailed analysis and discussions of the task/story, all estimators/team asked to select one card to estimate a user story. All estimators/team picks an equivalent value from the poker cards, which becomes the ultimate estimate. If values are different, then the moderators ask an explanation about estimators why they have provided values (taken into account

of high and low values). Discuss further on the requirement until a consensus achieved within the group. Planning Poker produces relative size estimates by grouping the similar-sized items from PBI.
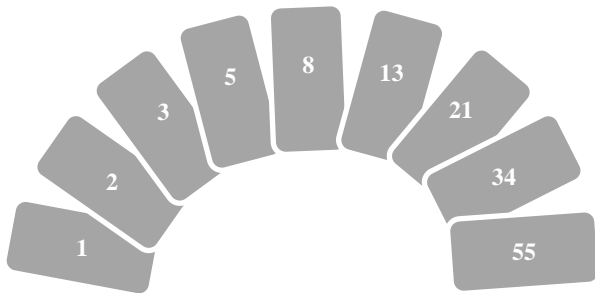

Figure 1: Planning Poker

3.2 T-Shirt Size

T-shirt model uses the sizes of XS (Extra Small), S (Small), M (Medium), L (Large), XL (Extra Large). The same way in Agile estimation uses T-shirt sizes for doing rough estimates for the backlog of items is vast. The model helps to do rough estimations very quickly for project scope. Decide a relative size (mostly Medium) after mutual discussion in the teams and arrive at one collective agreement by the team members for estimations—the agreed values assigned to the requirement according to the relative size assigned to Medium size. The primary advantage of t-shirt sizes is to estimate quickly and start the project and become familiar with relative estimating in the team—range type estimation rather than an absolute number. Rough estimations can be agreed upon by the team quickly.

Figure 2: T-Shirt Size

3.3 Dot Voting

Dot Voting method is the ranking between highest priority story to lowest priority stories from Product Backlog Most Essential Tasks/Stories, which should take forward to start first. Scrum wallboard used to stick all user stories with the yellow stickers and their description Voting is conducted between Stakeholders in the Scrum Meeting for each of the requirements to identify the priority requirements. Each Stakeholder provides their vote for the requirement in one dot (positive/negative). Product Owner rearranges the product backlog items from the most votes one to the least voted one as per the number of dots received by the Story. After completing the priority, the product backlog divided into 3 groups called high, medium, and low priority groups. Take one group of backlog items and do the same exercise between the group and identify the order of priority of the product backlog group. The activity repeatedly performed until the final order achieved with the agreement of all stakeholders. By this order, the priority ones would start first by the team and complete as per the agreed period. This method is straightforward and fast and works effectively to assess a small number of stories (up to 8-10).
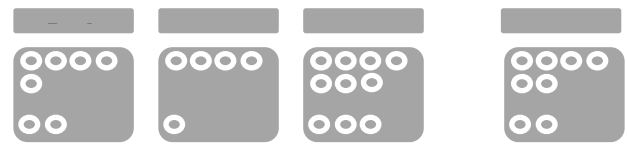

Figure 3: Dot Voting

3.4 Bucket System

Larger number of items and a larger number of team estimation is complex by using Planning Poker. Bucket System used for a larger number of items to be estimated by a larger number of teams. Various relative size buckets are created and assigned the numbers for each bucket from 0,1,2,3,4,5,8,13,20,30,50,100, 200. The estimator picks up the task/story from the backlog and place the Story into anyone the suitable bucket defined above. Proper explanation provides by the estimator of how that suites in that bucket. Use the divide-and-conquer technique to go the rest of the backlog items and assign a proper estimation bucket for each task/story. The activity repeated until the task/stories completed in the backlog. The Scrum Master roles it to check that nobody moves the task/story unless sanity checks are done. Bucket System is one of the collaborative estimation techniques and fast. Requires an experienced team to estimate the product backlogs into appropriate buckets.

Figure 4: Bucket System

3.5 Large / Uncertain / Small

A big way of estimation is the Large / Uncertain / Small method. This method is more simplified than the bucket system, where there are only three prearranged sizes rather than multiple buckets. The team or estimators estimate the item/story into any category called Large, Uncertain, and Small. This estimations technique used when there are similar items/stories in the Product Backlog. This method is similar to Simple / Medium / Complex estimations model. For each category, the number of Story points assigned for a Task/Story.

3.6 Ordering Method

This estimation technique is suitable when a large number of backlog items and a small number of resources. This method provides precise comparative sizes for the product backlog items. Low to High scale ranges are defined and placed them randomly on the priority ranges. Each team member or estimator asked to move the item from one scale to another. The order of the items/stories prioritized to move up or move down for one place until all participants in the team are satisfied with the Ordering of the product backlog from high to low. This activity provides the product backlog priority order.

3.7 Divide Until Maximum Size or Less

The Scrum team defines base value as the maximum size of the item for estimation (e.g., 8 hours of effort) in the project. Each product backlog is discussed in the grooming session to determine if the Story is relatively the estimated base value or less. If the Story is higher than the maximum size defined, it breaks the Story into multiple stories to support the estimated base value size. The activity continues until all the product backlog items are in the allowed size range.

The above Agile Scrum estimation techniques discussed are collective sizing and relative in size. All the stack holders in the project are participating in the estimation process. Most estimations derived in Scrum Estimation and Grooming sessions. The Scrum Master picks up the stories from the product backlog and discuss the requirement in the scrum meeting and estimates based on any one of the techniques described above. The collaborative decision taken for estimating the Story so that no one blames someone for inappropriate estimates. All the Agile estimation techniques described above used for relative units. User Story Points instead of the number of days/hours estimation. User Story Points gives the advantage of comparing each other regarding velocity. Velocity is a measure of work completed by the team in a particular sprint cycle. Based on the project scope, resource experience, and size of the project, a proper estimation technique has to adopt by the Scrum Master. The table-1 provides the details of when to use which estimations technique.

Table 1. Estimation Techniques

Estimation Technique	Small Number of Backlogs	Larger Number of Backlogs	Numerical Estimations	New to Agile Estimations	Long Term Plan Estimations
Planning Poker	Yes			Yes	
T-Shirt Size			Yes		
Dot Voting	Yes	Yes			
Bucket System		Yes		Yes	Yes
Large / Uncertain / Small		Yes	Yes		
Ordering Method	Yes		Yes		
Divide Until maximum Size or Less					Yes

While doing the story point estimations, conceptual

challenges came up in the team. Beware of the challenges in Agile scrum estimates; otherwise, it leads to confusion about how agile works. Accurate estimations from the team on the first day are a big mistake. Without proper coaching and experience, no team can provide story point estimates accurately. Training is one of the factors on how to estimate story points and let the team experience for one to two sprints. Lack of participation in scrum estimations leads to challenge in adopting agile. Sometimes team members inflated the story points, which may affect another resource in the learning phase. Agile provides the option to update the requirement, design, and architecture regularly to fit the business requirements. The story points associated with a story are abstract in value, and they differ from team to team. The team gets better at understanding its throughput capacity for predicting how much work they can realistically get done in the next Sprint. Agile encourages the concept of retrospective meetings, and the teams have to discuss how the Sprint went well and adjust the story points according to the team experienced in the past Sprint. Below table-2 describes the Agile Scrum Estimation challenges and their resolutions.

Table 2. Agile Scrum Estimation Challenges

Challenges	Resolution
Lack of story point estimation prediction	Teams do not know how to predict the story point from the backlog item. Management has to conduct considerable training programmed to the team how to work with Agile and do story point estimations.
No standardize estimations technique	Agile Scrum estimates are relative in size and not defined any size. The definition of story point is different from project to project. Scrum Master responsibility to adopt proper estimation techniques for their project size and experience of the team.
Lack of metrics like size, effort, and velocity from another project	Most of the teams do not maintain the data like size, efforts, and velocity of the project implementation if the data is available for other projects to estimate the same work type. Management has to support the team to use Agile Management tools to log the activities for further use.
Overestimation of Sprint if already not done previously the same type of work	Most of the time, developers inflate the story points if the team is working on new technology. Scrum Master responsibility to moderate the team and provide the relevant information on new technology to the team, not to inflate the stories.
Cultural and communication bandwidth	How the resource has understood the requirement is a primary factor for Agile delivery. Scrum Master responsibility to conduct

	daily standup meetings with the Product Owner for clearly understating of the requirement to be delivered on time.
Too big projects or backlogs	Split the backlog to as much as possible to fit into a single story point. Smaller tasks provide the team to get more clarity on requirements and deliver the item with the time frame. Create more small teams and work individually to deliver the common goal. Create Scrum-of-Scrums / SAFe teams to deliver large projects.
Non commitment of the resources	Commitment can be addressed by creating a self-managing and self-organizing team that allows the team to be creative, innovative, and acknowledged for their expertise.
Project Management Integration	Agile process is not suitable for projects to require a plan-driven approach. Create a hybrid Agile approach to integrate Project Management aspects into plan-driven projects.

4. CONCLUSION AND FEATURE WORK

Organizations are working on various popular Scrum Story Point estimations techniques used for the last decade. Various Organization adopts different estimation models according to customer goals. Scrum framework does not impose to use any estimations techniques. For example, Planning Poker and Story Points are not part of the Scrum framework. Scrum framework only creates rules of the game around estimates and gives the teams freedom to choose the estimation technique. Therefore, the teams can choose any estimation techniques described in Section-II depending upon their resource experience and required size. Organizations are adopting Agile very rapidly, and further work has to carry out to arrive at better effort estimation techniques to overcome the challenges in existing Agile Scrum Story Point estimates.

5. REFERENCES

[1] Adnan, M., & Afzal, M. (2017). Ontology based multiagent effort estimation system for scrum agile method. *IEEE Access*, 5, 25993-26005.

[2] Ahmed, M., Malik, B. H., Tahir, R. M., Perveen, S., Alvi, R. I., Rehmat, A., ... & Asghar, M. (2018, July). Estimation of Risks in Scrum Using Agile Software Development. In *International Conference on Applied Human Factors and Ergonomics* (pp. 111-121). Springer, Cham

[3] Altaleb, A., & Gravell, A. (2019). An empirical investigation of effort estimation in mobile apps using agile development process. *Journal of Software*, *14*(8), 356-369

[4] Arifin, H. H., Daengdej, J., & Khanh, N. T. (2017, March). An Empirical Study of Effort-Size and Effort-Time in Expert-Based Estimations. In *2017 8th International Workshop on Empirical Software Engineering in Practice (IWESEP)* (pp. 35-40). IEEE

[5] Bik, N., Lucassen, G., & Brinkkemper, S. (2017, September). A reference method for user story requirements in agile systems development. In *2017 IEEE 25th International Requirements Engineering Conference Workshops (REW)* (pp. 292-298). IEEE.

[6] Brizard, T. (2015). Challenges with Estimations. In *Broken Agile* (pp. 65-69). Apress, Berkeley, CA.

[7] Canedo, E. D., & Costa, R. P. D. (2018). Methods and metrics for estimating and planning agile software projects. *Twenty-fourth Americas Conference on Information Systems, New Orleans, 2018*

[8] Choetkiertikul, M., Dam, H. K., Tran, T., Pham, T., Ghose, A., & Menzies, T. (2018). A deep learning model for estimating story points. *IEEE Transactions on Software Engineering*, *45*(7), 637-656.

[9] Gandomani, T. J., Faraji, H., & Radnejad, M. Planning Poker in cost estimation in Agile methods: Averaging Vs. Consensus. In *2019 5th Conference on Knowledge Based Engineering and Innovation (KBEI)* (pp. 066-071). IEEE

[10] Hacaloglu, T., & Demirors, O. (2019, August). Measureability of functional size in Agile software projects: Multiple case studies with COSMIC FSM. In *2019 45th Euromicro Conference on Software Engineering and Advanced Applications (SEAA)* (pp. 204-211). IEEE.

[11] Hannay, J. E., Benestad, H. C., & Strand, K. (2018). Agile Uncertainty Assessment: For Benefit Points and Story Points. *IEEE Software*.

[12] Hasan & Khan. (2019). Software Development Methods – Properties and Advances. *International Journal of Computer Applications Volume 178 – No. 53*

[13] Hoda, R., Salleh, N., & Grundy, J. (2018). The rise and evolution of agile software development. *IEEE software*, *35*(5), 58-63

[14] Ibrahim, K. S. K., Yahaya, J., Mansor, Z., & Deraman, A. (2019, July). The Emergence of Agile Maintenance: A Preliminary Study. In *2019 International Conference on Electrical Engineering and Informatics (ICEEI)* (pp. 146-151). IEEE.

[15] Iqbal, J., Omar, M., & Yasin, A. (2019). The impact of agile methodologies and cost management success factors: an empirical study. *Baghdad Science Journal*, *16*(2), 496-504.

[16] Kaur, A., & Kaur, K. (2019). A COSMIC function points based test effort estimation model for mobile applications. *Journal of King Saud University-Computer and Information Sciences*.

[17] Khatri, S. K., Malhotra, S., & Johri, P. (2016, September). Use case point estimation technique in software development. In *2016 5th international conference on reliability, infocom technologies and optimization (trends and future directions)(ICRITO)* (pp. 123-128). IEEE.

[18] Moyo, S., & Mnkandla, E. (2020). A Novel Lightweight Solo Software Development Methodology With Optimum Security Practices. *IEEE Access*, *8*, 33735-33747.

[19] Munialo, S. W., & Muketha, G. M. (2016). A review of agile software effort estimation methods. International Journal of Computer Applications Technology and Research Volume 5–Issue 9, 612-618, 2016

[20] Prakash, B., & Viswanathan, V. (2017). A,"Survey on

Software Estimation Techniques in Traditional and Agile Development Models". *Indonesian Journal of Electrical Engineering and Computer Science*, 7(3), 867-876.

[21] Raunak, M. S., & Binkley, D. (2017, September). Agile and other trends in software engineering. In *2017 IEEE 28th Annual Software Technology Conference (STC)* (pp. 1-7). IEEE.

[22] Romero-Chacón, V., Muir-Camacho, H., Rodríguez-González, J., Gómez-Blanco, A., & Chacón-Rivas, M. Adapting SCRUM Methodology to Develop Accessible Web Sites. In *2019 International Conference on Inclusive Technologies and Education (CONTIE)* (pp. 112-1124). IEEE.

[23] Sarwar, A., Hafeez, Y., Hussain, S., & Yang, S. (2020). Towards Taxonomical-Based Situational Model to Improve the Quality of Agile Distributed Teams. *IEEE Access*, *8*, 6812-6826.

[24] Shams, A., Bohm, S., Winzer, P., & Dorner, R. (2019, July). App Cost Estimation: Evaluating Agile Environments. In *2019 IEEE 21st Conference on Business Informatics (CBI)* (Vol. 1, pp. 383-390). IEEE.

[25] Shimoda, A., & Yaguchi, K. (2017, July). A Method of Setting the Order of User Story Development of an Agile-Waterfall Hybrid Method by Focusing on Common Objects. In *2017 6th IIAI International Congress on Advanced Applied Informatics (IIAI-AAI)* (pp. 301-306). IEEE

[26] Takeuchi, H., & Nonaka, I. (1986). The new new product development game. *Harvard business review*, *64*(1), 137-146.

[27] The Standish Group 2018 Chaos Report. URL: https://vitalitychicago.com/blog/agile-projects-are-more-successful-traditional-projects/ Retrieved on April 04, 2020

[28] Yadav, A., & Sharma, A. (2018, May). Function Point Based Estimation of Effort and Cost in Agile Software Development. In *Proceedings of 3rd International Conference on Internet of Things and Connected Technologies (ICIoTCT)* (pp. 26-27).