ELSEVIER

Scopus

doi

Journal of Vibration Engineering

ISSN:1004-4523

Registered

- SCOPUS
- GOOGLE SCHOLAR
- DIGITAL OBJECT IDENTIFIER (DOI)
- IMPACT FACTOR 6.1

Our Website
www.jove.science

VIT Vidyalankar Institute of Technology
Accredited A+ by NAAC

Department Of Computer Engineering

DISSERTATION- I

SYNOPSIS

On

"***Agile-scrum study for astronomically immense-scale projects and ecumenically distributed software development projects to improve SDLC***"

Submitted in partial fulfillment of the requirements for the award of Degree of

Master of Engineering

Computer Engineering Department

By

Rushit. D. Davda

Under the guidance of

Prof. Suja Jayachandran

VIT Vidyalankar Institute of Technology
Accredited A+ by NAAC

DEPARTMENT OF COMPUTER ENGINEERING

VIDYALANKAR INSTITUTE OF TECHNOLOGY,

Wadala (E), Mumbai-37.

Academic Year 2020 – 2021

VIT Vidyalankar Institute of Technology
Accredited A+ by NAAC

Department Of Computer Engineering

Vidyalankar Institute Of Technology

University Of Mumbai

VIT Vidyalankar Institute of Technology
Accredited A+ by NAAC

Synopsis of Dissertation Work

On

COURSE	**M.E (Computer Engineering)**
NAME OF THE STUDENT	Rushit Dinesh Davda
SEMESTER	III
UNIVERSITY REGISTRATION NUMBER/PRN NUMBER	2019016402321092
DATE OF REGISTRATION/ PRN NUMBER GENERATED	
NAME OF THE GUIDE	Prof. Suja Jayachandran
PG TEACHER APPROVAL LETTER NO.	PG/ICD/2019-20/1898

"Agile-scrum study for astronomically immense-scale projects and ecumenically distributed software development projects to improve SDLC"

Signature of Student

Rushit Davda

Guide Signature

Prof. Suja Jayachandran

Abstract

Agile project management is a methodology that is commonly used to deliver complex projects due to its adaptiveness. It emphasizes collaboration, flexibility, continuous improvement and high quality results. Despite the wide adoption of agile methodologies, software development teams still struggle to meet requirements, partially due to practitioners' lack of motivation to apply agile techniques in practice. The Purpose of these methods is to make Scrum techniques more engaging & useful for practitioners. Program is employed on two batches with content analysis and survey results as benchmarks. A prototype was developed as a Jira/Trello Software app and evaluated with a real-world Scrum team. The study identified several positive effects from the good quality software produced, also some issues that need to be addressed, particularly on time management and team management skills, especially communication transparency. Employment of Agile Methodology improved client interest and involvement. This can positively impact practitioners' motivation by changing the atmosphere within the team, even if it does not contribute to the improvement of Scrum practices adoption. Agile-scrum study for astronomically immense-scale projects and ecumenically distributed software development projects to improve business change, limit customer feedback, scrum adoption and timely delivery. The research could help improve business change, limit customer feedback, scrum adoption and timely delivery.

1. Introduction

The main focus of agile methodology is customer satisfaction through continuous delivery. The use of agile methods creates high quality products and environments. In software development, scrum is an iterative methodology that depends on agile principles included in the Agile Manifesto. Agile software development emerged as a flexible, responsive, and team-empowering response to traditional software development and project management, where teams are intended to produce working software during short iterations.

There is evidence that agile projects are twice more likely to succeed and one-third less likely to fail when compared against a traditional methodology. Additionally, agile software development is associated with greater satisfaction than more traditional approaches.

In this research, efforts have been made to address the following questions:

Question 1: What are the influencing factors for the applicability of agile scrum methodology from the perspective of the software industry?

Question 2: Does there any framework exist in the literature for the applicability of agile scrum methodology?

To describe the solution and to develop the software faster and more precisely, and to make the result to be easily developed and maintained, the development of the software needs a specific methodology. Software development methodology is a process of organizing a set of methods and notation conventions which has been defined to develop software. SDLC is a software development life cycle that consists of several key stages in building software in terms of its development.

With the SDLC cycle, the process of building the system is divided into several steps and on large systems, each step is done by different teams. SDLC is not only important for the software production process, but also very important for the software maintenance process.

The results suggest that the metrics corresponding to Scrum techniques slightly increased after using the app when compared to the baseline, but probably this result cannot be attributed to practitioners' improved motivation.

Team members mentioned that the proposal should be more challenging, something that could be easily achieved by allowing organizations to adapt the rewarded behaviours.

In our study below we will be focusing more below as project management tools:

1. Jira
2. Trello

Jira is a project tracking software created for implementing the agile practices of project management. Jira is defined as an agile software that supports the scrum or Kanban agile methodology. Jira was mainly created for serving the purpose of project management and bug tracking in the software development field. Jira is also used as a ticketing tool during software development for tracking the errors and bugs related to the software and mobile apps. It helps to manage, plan, and monitor the agile software development practices. Jira is used by the software developers as an issue tracking tool which is used to organize, monitor, release, and implement new features in the software and for prioritizing the bugs.

Benefits of Using Agile Jira:

- Enhanced Visibility
- Prioritization of Task
- Enhanced Productivity and Efficiency
- Version Management
- Reviews and Retrospection

Trello excels in several departments. It's easy to use and has a simple interface that functions as a traditional whiteboard, only that it's hosted online. Its features aren't complex, meaning it doesn't occupy or obstruct the team from doing the work that matters. Trello is also very visual, allowing you to quickly see the stages of the project, the roles of team members, and the due dates of tasks. Instead of wasting time trying to interpret the details of a project, Trello's visibility allows you to focus on solving complex problems and completing tasks.

Benefits of Using Agile Trello:

- Ease of Setup
- Developer Friendliness
- Support for basic requirements of engineering project management
- Wide availability of integrations

2. Literature Survey

In recent years of software development there is an increasing interest in project managers to implement agile in the projects. Agile software development is an iterative way of developing a project. In agile the product owner communicates with stakeholders to get inputs and make a list of epics which will further be broken down into stories [2]. The stories are then prioritized and broken down into task. All the prioritize tasks will be maintained in a backlog. Agile is a group of waterfall models following incremental model [8].

In a rapidly changing economy, Business Intelligence solutions have to become more agile. Will help in creating an agile BI solution. Scrum is the most widely used agile development framework [1]. It guides the development process with its ability to create customer-valued software artifacts iteratively and incrementally. Measuring success in Scrum can be a challenging endeavor, with Power BI & other Business Intelligence (BI) tools we will be solving the real life problems. With the data there will be enormous help to the business in decision making. Flexibility and change adoption are key attributes for service-oriented architecture (SOA) and agile software development processes [4]. Study is designed to analyze the two diverse software architectural framework and development approaches, that is, SOA and Scrum process model. Also analyzes commonalities among Scrum process model and SOA architectural framework to identify compatibility between Scrum and SOA so that the Scrum process can be constructively used for SOA based projects. For

this purpose, an SOA based research and development project is selected as a case study using Scrum as the software development process [3].

Agile software development has the following attributes. It is incremental, cooperative and adaptive. Incremental here refers to the small release which tea delivers at the end of sprints. It is cooperative because there is always close interaction between teams and customers [1]. Finally it is adaptive because it has the ability to accept changes anytime during development. This concentrates on scrum implementation in different projects. Nowadays many project managers are choosing scrum agile over waterfall. But not all the projects are seeing success. Many of them fail because scrum agile doesn't fit the type of project they chose [7]. Many project managers are choosing agile scrum because of cross functional teams, fast paced environment and aggressive approach. I would like to put forward problems and experiences from several projects implemented by experts [2].

It is not just easy to bring a change in an organization or a team overnight. There are few things that should be taken into consideration before going agile. Going agile means not just having a scrum sized team, a product owner and a scrum master. It needs a lot of research performed before adopting agile. There are scenarios where various teams have implemented agile in their organizations and have failed to accomplish their project goals [4]. This is not because they don't have resources in required expertise but they have failed because they couldn't choose the right resources who can fit into an agile environment. Agile teams are cross functional which means you will have to be flexible to work with different sets of technical and functional people who don't have any clue what you are into. The way you collaborate is the key factor to success in agile scrum. This would be the problem I would be dealing with in this research paper. In a gist at the end of this research paper you will have an idea what could be the best ways of migrating to scrum.

3. Problem Statement & Objectives

The starting point for the artefact development was recognizing the link between poor adoption of Scrum practices and practitioners' lack of motivation to apply those practices, which is a subject of human behaviour. We then proposed this solution to promote a change of these behaviours in order to achieve the desired outcomes, and studied other works applying in software development to understand their strengths and gaps. With this, we derived more concrete goals based on Scrum's specification and the challenges affecting Scrum implementation identified through an analysis of related work and a questionnaire, whose methods are explained in the following subsections. Nowadays, as most of the project managers want to adopt agile scrum there are few things which they should consider before choosing the right approach for their project. This paper will give a few good inputs which would help them understand the success and failure scenarios of agile. This paper would help like a reference to scrum teams, project managers and other stakeholders who are interested in going agile.

In the scrum the scrum master focuses more on the development team process which means how to track, optimize performance and track the project progress. For this scrum master arranges daily base meeting sessions less than 30 minutes to get updates from team members. This daily meeting session is also a limitation in scrum methodology. Because as mentioned that scrum plans the sprint work, effort and task etc. to team members but it is difficult to find the right person in the team for the right task. In the scrum mostly the team members take the task with their own plan often not discuss with the other team members. In this case when the developer does not produce the desired results and is less coordinated with the team member/scrum meeting then the scrum master decides to supervise him that ultimately creates more issues. The developer then does not participate in the daily meeting session instead he was seen as an expert in the current sprint. The scrum master also faces that the team is not responding to him accurately. The developers are not giving a right update of a sprint to scrum master due to this scrum master has

lost trust on the team. Due to the daily meeting session the all team members become agitated and feel pressured from the scrum master to complete their task. These daily scrum meeting issues have an awful influence on the project. The developer does not want to attend the daily meeting session and give their work update to other developers in the team.

Therefore, scrum is not applicable for the large-scale organization where the team members are greater than the agile team, projects are more complex and larger and team members are not used to daily meeting session. It is also difficult to arrange meeting sessions in the large-scale organization due to their geographical and distributed software development nature because it will be difficult together with all the developers at the same time.

The toughest thing about Agile is to help teams internalize cultural changes be they a small change such as teams coming to meetings on time or transformation such as creating high performing self-organized teams. As we all know, Agile manifesto and values emphasize more on human elements. It is not easy to change behaviors. If I have been programming without writing a single test, asking me to do TDD or BDD is going to be tough.

a. Existing System

1 INTRODUCTION

Software development process is basically used to de- velop software products which follow specific processes. Software development process is also known as software process life-cycle (SDLC). It comes into the category of soft- ware development life cycle. There are number of models with variety of tasks and activities, Agile is one of them

1.1 Agile

Agile software development is an iterative ,incremental and time-boxed approach which comes under a group of software development methods where requirements and solu- tions evolve through collaboration between self-organizing and cross-functional teams. It promotes adaptive planning and conceptual framework that promotes foreseen interactions throughout the development cycle.

1.2 Scrum

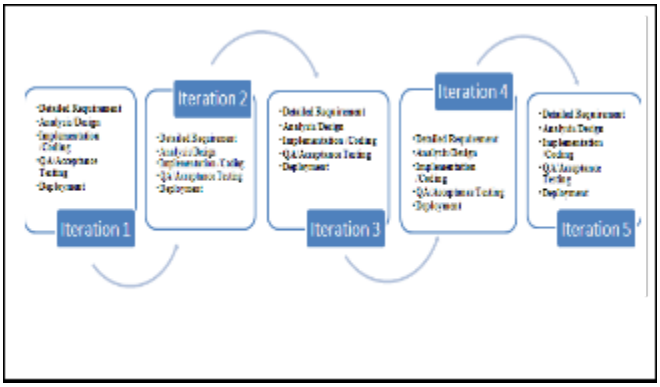


Fig 1 : Iterative and incremental approach
Ref: scrum.org

- Scrum is a process used for agile software development basi- cally it is a framework rather than full process or methodology. So software development team is responsible to decide detailed description “how everything is to be done in the project” .In scrum, a series of iteration is known as a sprint which decides the progress and time boxing for the project. Scrum is daily used for those projects which have rapid changes and highly emergent requirements.

2 SCRUM ROLES

There are many key persons with different roles and responsi- bilities who give their guidance, which make scrum model best from other models.

2.1 Product owner:

- Single person responsible for maximizing the develop- ment effort.
- Responsible to give new visionary direction in development of product.
- Design product backlog list which is indirectly “the voice of customer” and prioritizes the user story in product Back- log by ensuring the value of the work the Development Team does.
- Final approval and stamping of requirements questions is done by the product owner.
- Decide the acceptance or rejection for each product increment which may be gives the direction to continue or end the development process.
- Take decision regarding product “shippable mode”.
- He/she has a role of leader who may be a member of the development team but we cannot combine his/her role with scrum master.

2.2 Scrum master:

- Scrum master works as team’s coach who helps the team members to achieve their goals with high performance.
- Scrum master is not traditional project manager because he/she works as project manager com as developer with team members.
- Scrum master gives the direction to team members by this they can only focus on the goal which they have selected during each sprint.
- Helps the team members if they get any obstacles.
- Creates a favorable environment for the team’ self- organization
- He/ She Has a leadership role over team members.

2.3 Scrum development team:

- The Scrum development team has two functions one is Cross-functionality and second is self organizing.
- Self-organizing / self-managing, without externally assigned roles.

- Cross functionality in scrum teams enables team members to do different kinds of work in a team.
- Scrum development team members come with various job titles like (Developer, Tester Executives, Customer) Each team member contributes their best they can do for each sprint.
- Negotiates commitments with the Product Owner.
- Average Size of development team is recommended.

3. SCRUM MEETINGS:

3.1 Sprint planning meetings:

- The product owner and team member along with the scrum master conduct a sprint planning meeting for de- ciding which product backlog item they would take to convert, working product in each sprint.
- Product owner is responsible for prioritizing the items which are more important.
- Daily scrum and sprint execution: The team members are free to select a balanced amount of work which they can implement without any technical obstacles.
- The team member’s pull work from the product backlog and complete it in the sprint.

3.2 Sprint review Meeting:

- Sprint review meetings only demonstrate an incremental working product to the product owner and everyone else.

- Live demonstration is necessary rather than a report.
- End of the sprint review meeting, the product owner reviewed and revised the sprint planning meeting.
- The scrum master helps the product owner and stakeholders to revise the product backlog items with new priorities.
- It is like an inspection meeting for external stakeholders (even end users) to attend and propose or refine the re- quirements.

3.3 Sprint Retrospective Meeting:

- Sprint retrospective meeting is conducted after comple- tion of each sprint
- The team member generates and reflects on its own process during this they can inspect their behavior and take appropriate action for future sprint.
- Three things are going to be noticed :

  1) Is everything going well?

  2) Need any improvement?

  3) Stop, if problem occurred?

3.4 Daily scrum meeting:

- Scrum master and development team members append a total of 15 minutes at the same time and place to re- porting to each other.
- Three questions are generated in this sprint during sprint execution and some common addition fetchers are detected to achieve the sprint goal.

3.5 Backlog Refinement Meeting:

- The product backlog which is decided by the product owner is further analyzed at granular level and certain refinement added into it which make it more helpful in developing a release backlog.

4 SCRUM ARTIFACTS:

- Scrum artifacts are the various documents that are created before or during the sprint in the project.

4.1 Product backlog:

- Prioritize ranked list with desired functionality and set of user story which is atomic unit of work.
- Priorities of the user stories are decided by the product owner.
- Product owners can re-prioritize the list during or at the end of the sprint.

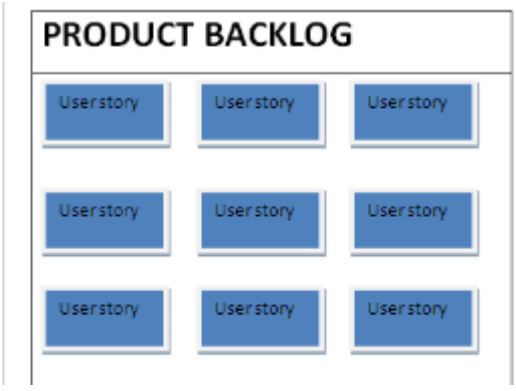


Fig. 2 Product Backlog (set of User Stories)
Ref: scrum.org

- It is transparent to all stakeholders.

4.2 Product backlog items:

- Product Backlog Items are nothing just as a user story.
- Product Backlog Items define the product widely and prevent it from technical fault.
- Effort estimation is calculated by the relative unit that is the story point.

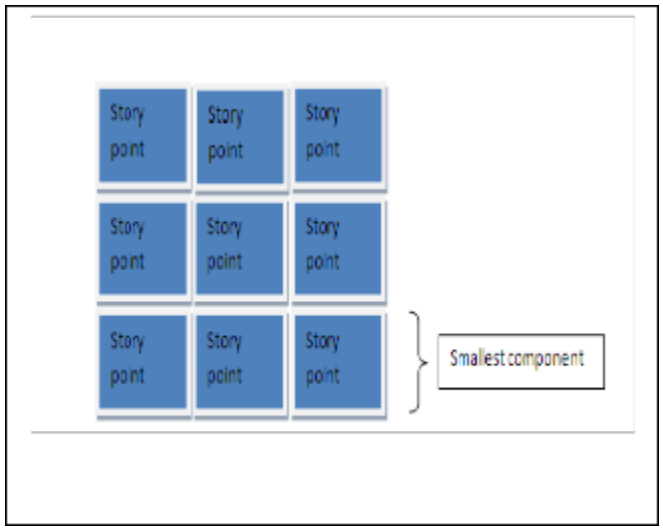


Fig. 3 Collection of story point comes out as smallest component
Ref: scrum.org

4.5 Burn down chart:

- Signifies the total remaining work hours in one sprint.
- It is estimated day by day, which may go up or go down.

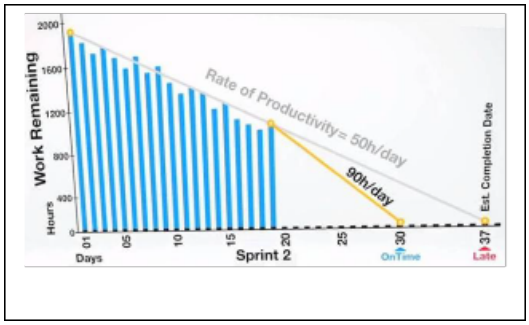
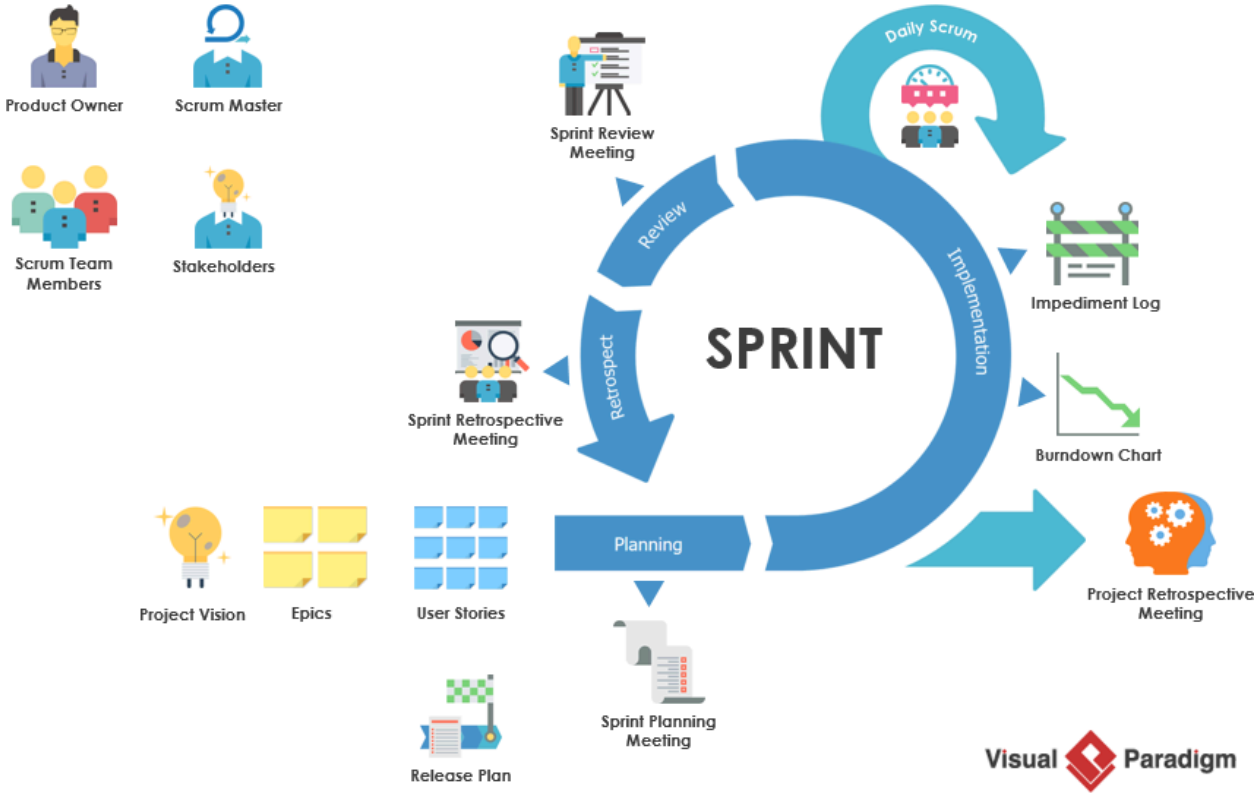


Fig. 5 Burn down chart
Ref: scrum.org

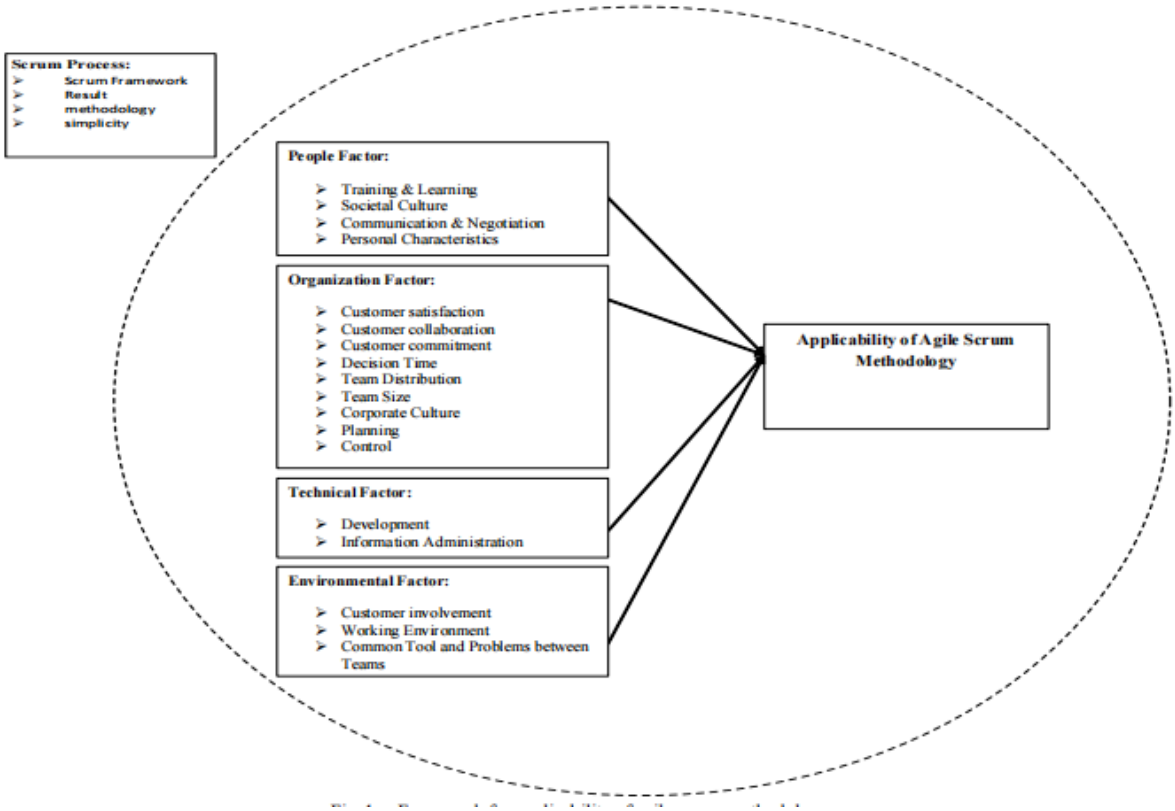
- Team gets self organized on the basis of day by day work.

4.3 Sprint backlog

- Product backlog items are decided or negotiated be- tween the team members and by product owner, dur- ing sprint planning meetings.
- Requirements should not be changed during sprint ex- ecution.
- Initially team members defined the task during the sprint planning meeting.
- Transparent to all team members and it can be revised



Agile Scrum Framework
Ref: scrum.org



Framework for applicability of agile scrum methodology

Ref: scrum.org

b. Proposed System

After identification of critical factors and extraction of key factors as discussed in the previous section, the next step was to develop a framework. The key factors were analyzed in detail and four factors were selected to develop the framework. The selected key factors finally included in the proposed framework are people, organizational, technical and environmental.

1. Define your target behaviors (based on business goal)

What behaviors as a team do you want?

What activities can we simulate?

In our scenario, when started, the organization said that they have been doing agile i.e doing all agile ceremonies, but you do not see the signs of agile teams ex. When you look at stand up boards nothing is updated, almost everyone is coming to meetings almost 15 min late

2. Build or customize as per the needs that you defined

There are many websites and tools available handy for the coaches to determine tools required.

For my problem of transformation, first we wanted to create a visibility of their teams. It is like the Olympics where each country's medals are tallied at the end of the day. The Board was displayed at the entrance and updated every day. Each team chose its own name and we had scoring on different parameters (defined as per target) such as timeliness or backlog quality. The teams that won got ribbons/plaques hanging in their team space to create a festive atmosphere. That also created friendly competition and collaboration with other teams.

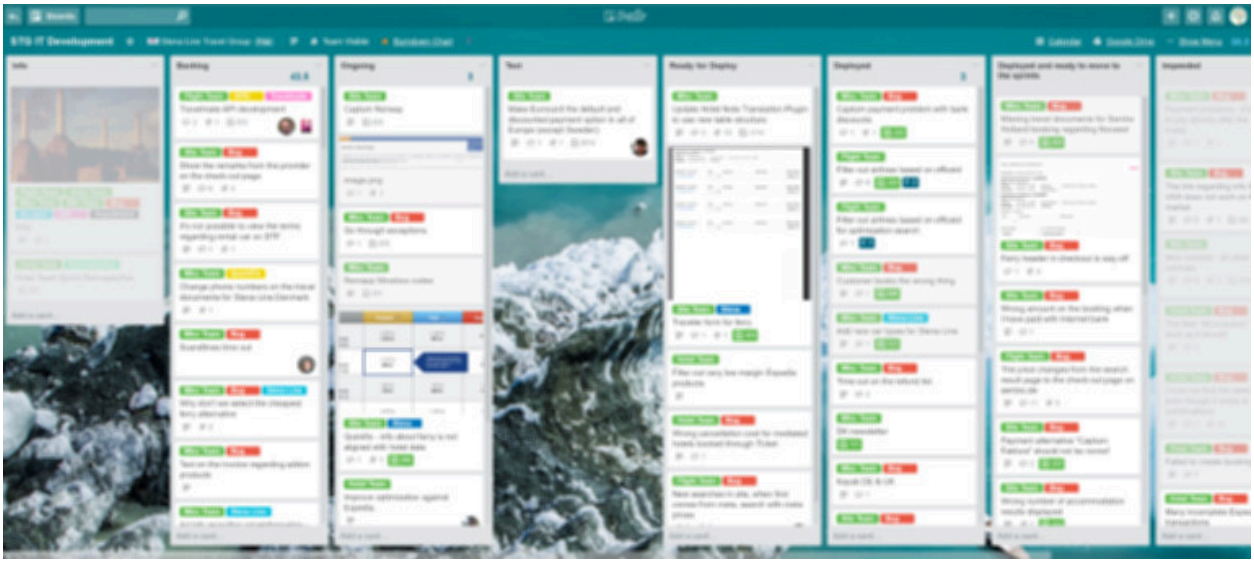
3. Run /Measure Progress

Behaviors are usually difficult to track. It is better to get inputs/feedback from the team on the progress. Use leaderboards, Widgets etc

4. The fourth step is Inspect and adapt

Lastly, create rewards (Both Intrinsic and extrinsic rewards) to motivate and sustain the momentum that is created. We have got better outcomes using this for agile transformation. Teams enjoy and start taking ownership as we

progress.



Our Scrum Trello board
My project Trello Board

4. Conclusion

- Based on the analysis that has been made and the results of the discussion in the previous section, then some conclusions can be drawn as follows:
    - The quality and project risks are seen more quickly.
    - Business Change and Customer Feedback will be visible in the final sprint.
    - Speed of visible projects in real time.
    - Live Application can be done per product backlog.
- In future studies a new method will be added to improve business change and customer feedback should be limited so that the project is not delayed.

5. References

[1]. *Mengze Zheng, Islam Zada, Sara Shahzad, Javed Iqbal, Muhammad Shafiq, Muhammad Zeeshan, and Amjad Ali (February 2021)*, Key Performance Indicators for the Integration of the Service-Oriented Architecture and Scrum Process Model for IOT.

[2]. *Simone Briatore, Alessandro Golkar (January 2021),* Estimating Task Efforts in Hardware Development Projects in a Scrum Context.

[3]. *Sikandar Ali (January 2021),* Analysis of Service-Oriented Architecture and Scrum Software Development Approach for IIoT

[4]. *Nesib Tekin, Mehmet Kosa, Murat Yilmaz, Paul Clarke, Vahid Garousi (August 2020),* Visualization, Monitoring and Control Techniques for Use in Scrum Software Development: An Analytic Hierarchy Process Approach

[5]. *Saikat Gochhait, Ayesha Khalid, Shariq Aziz Butt, Tauseef Jamal (June 2020),* Agile Scrum Issues at Large Scale Distributed Projects: Scrum Project Development At Large

[6]. *Giuseppe C. Ng (January 2019),* A Study of an Agile Methodology with Scrum Approach to the Filipino Company-Sponsored I.T. Capstone Program

[7]. *Yilmaz, M (2019),* Observed effects of software processes change in three software firms: Industrial exploratory case study.

[8]. *Mihaela Muntean, Traian Surcel (September 2013),* Agile BI - The future of BI