Enabling the
Digital Enterprise

Purple Paper

Project Development Methodology Selection

Arrk Group uses several methodologies for the management and governance of projects.

The selection method has historically been a question of good judgment, previous experience and the best fit with the customer requirements.

This document demonstrates the rationale by which we have generated an objective selection process for project methodology choice.

UK:
Greenheys, Pencroft Way,
Manchester Science Park,
Manchester, M15 6JJ

India:
Building 5, Sector 2,
Millenium Business Park,
Mahape. Navi Mumbai - 400 710

+44 (0) 161 227 9900
www.arrkgroup.com
talktous@arrkgroup.com

Key Considerations

- Quality
- Process & Governance
- Cost
- Business Deadlines
- Resourcing
- Customer Willingness

Overview

Arrk Group works with a wide range of customers, ranging from SMEs to multinational corporations with unique enterprise level requirements.

In the discovery period phase of any project, which we call EmbArrk, a standard series of questions are asked, aimed at determining the best project methodology.

Various agile methodologies epitomise the philosophy and intent as indicated in Agile Manifesto. From an implementation point of view, each methodology prescribes some specific practices to deliver software outcomes in an agile manner. Each methodology is different in terms of rigour, flexibility, areas of focus, level of preparedness needed and so on.

As part of our continuous development and refinement of best practices in the use of a range of development methodologies and practices, we have developed a matrix to not only consolidate our learnings on using an agile approach to development in distributed teams, but also to create a reference matrix to aid decision-making.

In developing our matrix, two project types were evaluated, a New Development is typically a ‘green field’ scenario where new software is being developed. The second scenario are Support projects which generally involve the maintenance and support of an existing application. Both types require a different mix of considerations, therefore selecting the most appropriate methodology is a challenging task requiring good alignment of what the methodology entails to the context and needs of the project.

Project Type Considerations

Matrix Questions

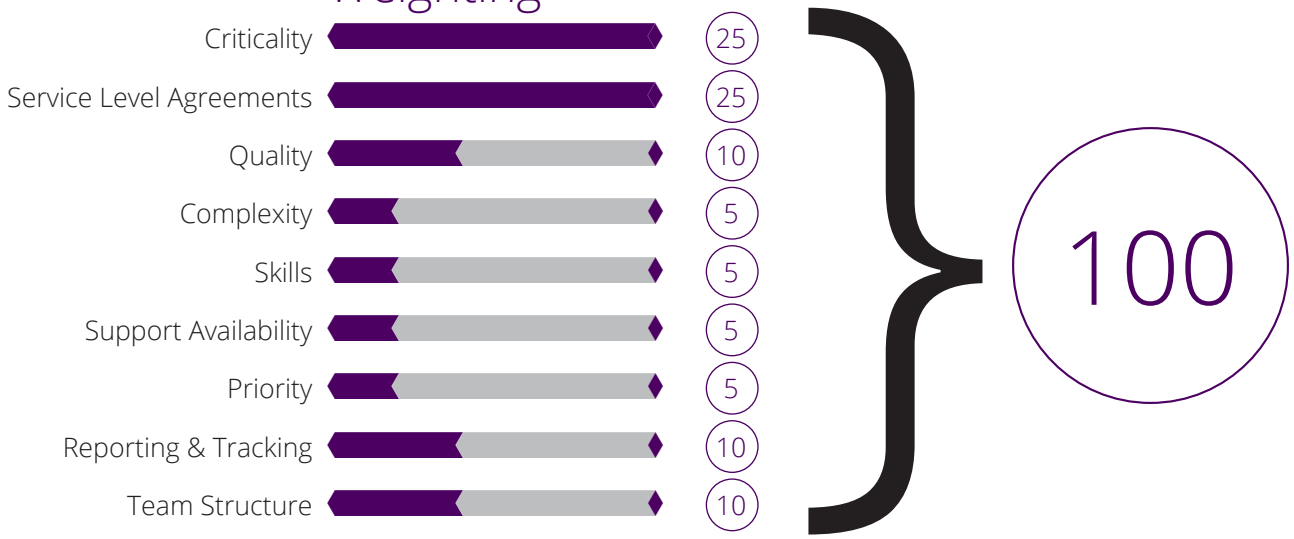
1: New Development Projects

Consideration	Description
Criticality	Whether the methodology can be used to develop software that is critical to the delivery of the organisation's services to its users.
Quality	Whether the optimal quality expectations can be met by execution of the methodology.
Complexity	Whether complexities of any and all types of processes can be handled through the execution of what the methodology entails.
Skills	Expectations that the methodology has about the team from a skills perspective: cross-functionality, specialist or both.
Reporting & Tracking	Optimal reporting as is needed for carrying out and tracking development work is supported from an end-deliverable perspective.
Dependency	Whether external dependencies can be managed and handled within the scope of the methodology.
Team Structure	What the methodology expects and how well it aligns with a team that is distributed or not, co-located or not, same time-zone or not.
Planning	Whether the adoption of the methodology would positively influence the planning efficacy.
Customer	Whether the customer has the willingness to embrace agile methodology.

2: Support, Maintenance & Enhancement Projects

Consideration	Description
Criticality	Whether the methodology can be used to support software that is critical to the delivery of the organisation's services to its users and respond to immediate fix requirements.
Service Level Agreements	Provides any mechanism to ensure compliance with Service Level Agreements (SLAs) or agreed timely turnaround.
Quality	Whether the optimal quality expectations can be met by executing the methodology.
Complexity	Whether complexities of any and all types can be suitably handled.
Skills	Requirements that the methodology has about the team from a skills perspective.
Support Availability	Whether availability of support personnel as and when required is considered.
Priority	Whether the work prioritisation is supported and to what extent.
Reporting & Tracking	Optimal reporting as is needed for efficiently carrying out and tracking support work is feasible.
Team Structure	What the methodology demands and how well it aligns that is distributed or not, co-located or not, same time-zone or not.

Weighting



Project Types

For the purposes of this matrix, we considered four popular Agile methodologies:

Kanban

The product of pioneering thinking at Toyota, Japan in the 1970s, Kanban (which translates from Japanese into English as signboard or billboard) is a scheduling system for just-in-time (JIT) production. Subsequently embraced by other disciplines, Kanban has become an effective tool in support of running an efficient production system. One of the key benefits of this methodology is to establish an upper limit to the work in progress inventory, which avoids system overload.

Scrum

Unlike a traditional, sequential approach to product development, Scrum is an Agile software development framework, where self-organising teams work as units to reach a common goal. A guiding principle of Scrum is recognition that customers change their requirements, and this methodology has the capability of meeting unpredictable challenges of requirements churn. Scrum adopts an empirical approach, with the team focusing on delivering quickly and responding to emerging requirements.

Scrumban

A portmanteau of Scrum and Kanban, Scrumban is a methodology based on the principles of both. Suitable for projects which feature unexpected work items, the methodology incorporates the daily meetings and practices of Scrum without the time-limited sprints. Meanwhile, visualisation of the work stages and limitations of simultaneous unfinished work are immediately familiar with that of Kanban. A team's work-flow is designed engage all members and keep to a minimum completion time for each work time.

Extreme Programming (XP)

An alternative type of agile software development, XP advocates frequent releases in short, sharp cycles. This extreme speed is intended to improve productivity and introduce checkpoints at which new customer requirements can be incorporated. Often incorporating the idea of pair programming, XP involves extensive code reviews, where one programmer codes while the other reviews. XP also features a flat management structure and the avoidance of programming features until they are needed.

Compliance Scale

Against each consideration we assign a compliance scale, measuring between 0 and 1, in increments of 0.25:

1

The principles of the methodology are clear and/or prescriptive to ensure its execution and that the project type consideration can be dealt with in a satisfactory and efficient manner.

0.75 0.5 0.25

Relative to other methodologies, or by itself the scores are indicative of the fact that the principles of the methodology are not clear and/or not prescriptive enough for the project type consideration to be carried out satisfactorily.

0

The methodology execution would render the project type consideration to be not achieved.

We then create a score for each methodology by summing the product of weight and degree of compliance.
This matrix contextualises what characterises each methodology for a project type, helping deepen its understanding and value.

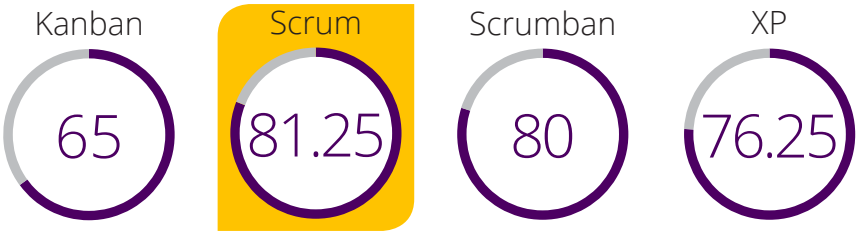
20 X 0.5 = 10

Matrices

New Project Development

Consideration	Weighting	Kanban		Scrum		Scrumban		XP	
		Compliance	Score	Compliance	Score	Compliance	Score	Compliance	Score
Criticality	10	1	10	0.5	5	1	10	0.5	5
Quality	20	0.5	10	0.75	15	0.75	15	1	20
Complexity	20	0.5	10	1	20	1	20	1	20
Skills	10	1	10	1	10	1	10	1	10
Reporting & Tracking	15	0.25	3.75	1	15	0.75	11.25	0.25	3.75
Dependency	5	1	5	1	5	1	5	1	5
Team Structure	10	1	10	0.5	5	0.5	5	0.5	5
Planning	5	0.25	1.25	1	5	0.5	2.5	1	5
Customer	5	1	5	0.25	1.25	0.25	1.25	0.5	2.5

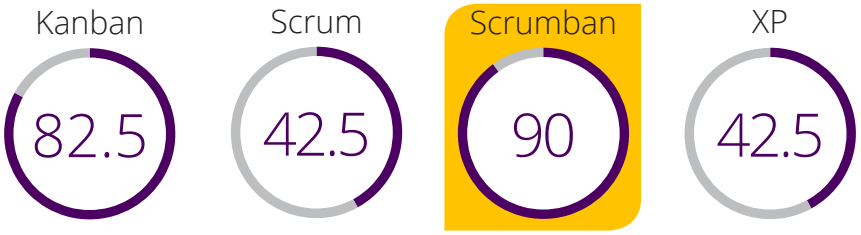
Overall Scores



Support Projects

Consideration	Weighting	Kanban		Scrum		Scrumban		XP	
		Compliance	Score	Compliance	Score	Compliance	Score	Compliance	Score
Criticality	25	1	25	0	0	1	25	0.5	12.5
SLAs	25	1	25	0	0	1	25	0	0
Quality	10	0.5	5	0.75	7.5	0.75	7.5	1	10
Complexity	5	0.5	2.5	1	5	1	5	1	5
Skills	5	0.5	2.5	1	5	1	5	1	5
Support Availability	5	1	5	1	5	1	5	1	5
Priority	5	1	5	1	5	1	5	1	5
Reporting & Tracking	10	0.25	2.5	1	10	0.75	7.5	0	0
Team Structure	10	1	10	0.5	5	0.5	5	0	0

Overall Scores



Conclusion

At Arrk Group, the use of the Reference Matrix helps guide the selection of project methodology or combinations of methodology to ensure we start right.

The outcomes of optimal methodology and their pecking order arrived at by the matrix can be deemed as quite logical given the project type context and what the methodology brings to the table.

For New Development projects, it is not surprising that Scrum represents the best fit because of:

- The level of rigour that Scrum provides related to planning and governance.
- The cross-skilled team (expected in a Scrum team) that can best handle the complexities of new development.
- The regular delivery of shippable software increments achieved for the customer within a framework of prioritised backlog complemented by progress tracking, ongoing visibility to customer and promise of continual improvement via ceremonies so typified by Scrum.

For Support projects, Scrumban scores the highest due to:

- It represents the best of Scrum rigour and Kanban flexibility.
- It being responsive or adaptive such that critical items can be dealt with to meet business demands without any delay. Service Level Agreement turnarounds are therefore more likely to be met along with the assurance of quality.

As much as the methodologies individually score in a particular context with reference to what is mentioned above, it must be borne in mind that a potent mix of two or more methodologies may suit a particular project's needs, and shying away from it or disregarding it would truly not be agile.

About Arrk Group

Arrk Group delivers high quality software engineering solutions via a collaborative, win-win partnership model.

Founded in 1998, we use leading software engineering practices, distributed teams and lean consulting services to deliver industrial strength digital and mobile solutions.

Get In Touch

+44 (0) 161 227 9900

www.arrkgroup.com

talktous@arrkgroup.com

UK:
Greenheys, Pencroft Way,
Manchester Science Park,
Manchester, M15 6JJ

India:
Building 5, Sector 2,
Millenium Business Park,
Mahape. Navi Mumbai - 400 710