Agile Working

Agile software development[1] describes an approach to project management under which requirements and solutions evolve through the collaborative effort of self-organizing and cross-functional teams and their customer(s)/end user(s). It advocates adaptive planning, evolutionary development, early delivery and continual improvement, and it encourages rapid and flexible response to change.

The terms "Agile" working, "Free Address," and "Hoteling" are often used interchangeably, creating confusion in workplace planning. This topic overview defines and explores Agile implementation and the impact of it on workplace planning. In addition to defining Agile, this brief will review appropriate workplace planning elements and provide tactics to reduce future risk and product suggestions for solutions. For further information, refer to the Free Address topic overview and the Knoll educational presentation.

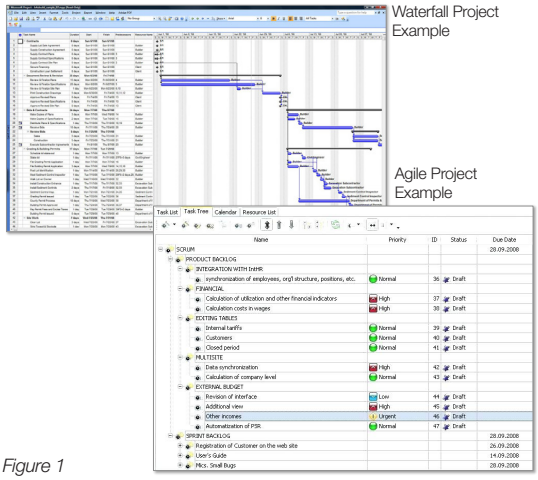


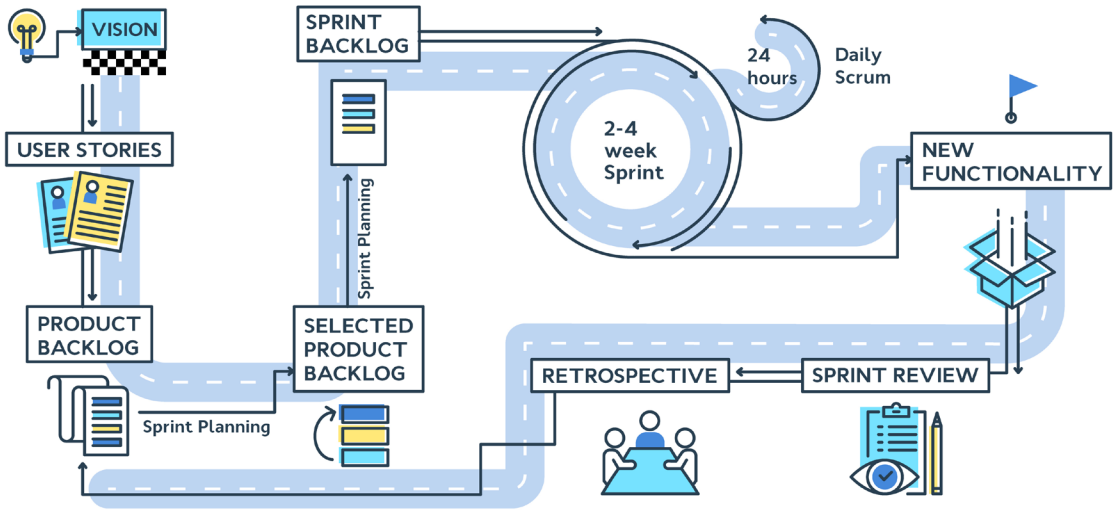
Figure 1

Common forms of Agile implementation:

Scrum: cross-functional groups comprised most often of a "Scrum Master," project manager, testers, quality assurance (QA), and several software developers.

Kanban: a "LEAN" methodology that seeks to address capacity along with demands, eliminating system-wide bottlenecks.[2]

Agile differs from traditional project management methodology, which is known as "Waterfall," due to the resulting shapes on a Gantt chart (see Figure 1). Waterfall projects focus on careful sequencing of dependent events and activities. Construction projects are "Waterfall" projects – floors must be finished before carpet installation, which happens prior to furniture installation, etc. Agile development is fundamentally different in that elements of the finished product are divided into discreet segments and work is achieved simultaneously.[1]



People

Agile work is most often implemented and organized utilizing “Scrum” teams – a term taken from the game of rugby whereby all players on the pitch work together as a group at the same time to get the ball and control its progress downfield. Agile scrum teams range in size from 8-24, and are comprised of software developers, a project manager, quality assurance (QA) personnel, testers, and are led by the Scrum Master. Scrum masters receive additional training and certification; it is their goal to coordinate all work, keeping an eye on capacity to ensure work milestones are being met on-time. Project managers are responsible for communicating work progress across the organization to relevant stakeholders and project owners.

Process

+ Agile approach allows for the simultaneous creation of multiple project elements, achieving faster delivery
+ All work is separated into small, achievable segments within a larger project, assigned to several Scrum teams.
+ Segments of work are further sub-divided into “sprints.” Duration of the sprint cycle can be as little as 14 days (2 weeks) or as long as 56 days (8 weeks). 4-6 weeks is common for on-going project/product work.

Deliverables from a sprint are “Released” during what is called “Go-Live” – when the new code is pushed out to the “live” customer-version of the software system (“customer" can also refer to internal users who may own enterprise-wide systems such as Finance, HR/Personnel management, etc). Go-Lives are scheduled when there is least customer use and demand, and usually requires a small “outage” when the system is down. Therefore, most “Go-Live” events are over weekends and overnight. When planning a furniture move and install, one will need to ask if the move will be impacting any Go-Live events as the disruption would be critical and negatively impact the release.

Planning

Planning spaces for Agile working requires answering a few key questions in order to begin identifying furniture type and configuration.

+ Size of typical Scrum Team
+ Sprint length
+ Type of work: multiple applications for multiple clients or internal systems development for a single customer/company
+ Technology: number of monitors, data management, project management tools
+ Frequency of team member changes
+ Maturity of Agile implementation

The average size of scrum teams will inform the cluster size(s) for your workstation configuration; sprint length will dictate when moves/personnel shuffling will occur – important for facility managers, dealer project managers, and installer teams. The type of work matters: internal system teams will change personnel less frequently. Teams working on project work for clients will change personnel often depending upon the project size and content, much like project teams at an A&D firm. Sprint length is important as that will impact internal planning of moves and target dates for facilities projects.

Technology: The number of network data drops may be more than 2 per user for software testers and QA personnel; extensive use of VPN may hinder mobility. Many developers on scrum teams may be located off-site, so headsets for clear communication and web cameras are essential, particularly across cultures. Regarding monitors, a minimum of two is standard and developers often prefer to spin one monitor to vertical to show maximum lines of code. Having this ability to turn the monitors is a key feature to consider when recommending and specifying monitor arms. Upon installation, it is important to ensure that there is sufficient space between the two screens and enough cable length to accommodate this behavior.

Maturity of Agile Implementation: understanding the stage of implementation will provide a guide to how often a scrum team will reconfigure/change members. Early Agile implementations will see much more frequent moves and personnel changes as an organization determines its best expression of Agile working. Once established (usually around 24 months/2.5 years), personnel shuffling will be greatly reduced, changing personnel very infrequently between major projects or, splitting a successful team to create an additional scrum team.

Sound management: Teams begin the day with a "Stand-Up," which is at most 15 minutes in duration, then are typically quiet while focused on stationary, heads-down work. Agile working hinges upon the ability to quickly respond to issues and adjust, so teams will frequently erupt into noisy discussion unexpectedly throughout the day. During planning sessions, brainstorming is critical to creative and adaptive problem-solving. Support this with plenty of markerboard surfaces for each team. A mix of fixed and mobile markerboards is ideal so that each team can keep some of their work desk-side.

Hospitality: Given the intense focus of development work, team members welcome opportunities to take breaks in comfortable environments away from their primary workspaces. Productive planning and problem-solving happens during these times, so the hospitality spaces should have plenty of surfaces for laptops and markerboards to facilitate discussion. Planned social time is an essential element in Agile culture so shared spaces are extremely important.

Agile in Free Address Environments?

Typically, Agile scrum work is not suited for free address environments because of the team-based nature of the work. Scrum teams need seating together and are typically in the office every day. Ideally, colocation of related teams supports idea-sharing and smooth transitions when personnel do move between teams. Product owners need to be accessible by the scrum teams, so free address environments may add time delay for connection and communication. Importantly, the strength of the Agile scrum model is the close-knit team dynamic; celebrating team spirit is essential.

Agile scrum teams typically develop team names, mascots, colors, and themes – incorporating and celebrating those team artifacts and symbols into the workspace supports their culture. Agile teams work intensely and need breaks for social activity and celebration of accomplishments. Creating a social committee to organize team events: ping-pong tournaments, pizza parties, cornhole tournaments, March Madness viewing, monthly birthday parties, etc, all support the truly special nature of successful scrum teams.

Free Address rarely supports Agile Scrum work because:

+ Scrum teams need seating together
+ Ideal to collocate related teams
+ Product owners are accessible
+ Celebrate team spirit!

• • •

1. The term Agile was popularized by the Manifesto for Agile Software Development – which one can read on Wikipedia. The values and principles espoused in this manifesto were derived from and underpin a broad range of software development frameworks, including Scrum and Kanban.

Wikipedia contributors. "Agile software development." Wikipedia, The Free Encyclopedia. Wikipedia, The Free Encyclopedia, 30 May. 2018. Web. 1 Jun. 2018.

2. LEAN manufacturing approach seeks to eliminate waste from every step or process. Types of waste are: transport, inventory, motion, waiting, overproduction, over processing, and defects.