SOFTWARE RELEASE PLANNING FOR AGILE PROJECT MANAGEMENT

Downer, Jane
Illinois Institute of Technology
jdowner@hawk.iit.edu
A20452471

Krishnan, Ramya
Illinois Institute of Technology
rkrishnan1@hawk.iit.edu
A20506653

Radhakrishnan, Swetha
Illinois Institute of Technology
sradhakrishnan@hawk.iit.edu
A20460652

I. Introduction

Planning the next product release is recognized to be one of the most challenging parts of market-driven product development and a critical success factor in agile software development projects. Finding an acceptable scope for a release while taking into consideration financial, resource, technical, and other constraints is the major objective of release planning.

In order to determine when a releasable version of a software product should be made accessible and what features should be included, a high-level strategy must be created. A release strategy will typically consider the next two releases. An organization can benefit greatly from a release plan. It can be used to direct development, evaluate its status, and support activities related to strategic planning.

Even the most experienced development teams frequently encounter difficulties while creating release plans. The ability to balance available resources with desired functionality and a strong technical understanding of the desired features are requirements for developers. The degree of uncertainty in the time and cost to develop the functionality of choice as well as in the expected value returned by the features further complicates the issue.

The most effective method for creating and delivering features during the software development cycle through sprints and releases is agile release planning.

II. Agile Release Planning

Agile release planning is a project management methodology in which teams develop and deliver products over the course of several iterations or small releases rather than all at once or in a series of major releases. Agile release planning allows teams and project managers to divide up a large, restrictive schedule into smaller, more manageable pieces of work that correspond to specified smaller or incremental feature releases.

The scheduling of each incremental release and the dates by which those releases should be finished is referred to as the planning component.

One of the popular project management approaches is agile project management, which includes agile release planning (also known as Scrum release planning). Teams break down work into brief sprints, often lasting two weeks, when using agile (and similar approaches like Scrum). To prevent getting lost in the pile of unfinished projects, teams benefit by breaking work down into these manageable segments.

A scheduled release typically includes the results of numerous sprints. The important point to remember is that each release will be smaller and gradual, and it can be changed as needed to meet the needs of the project.

A. Purpose

Customers are given early access to components of a system through incremental development, which gives them a sense of value and an early chance to offer input. Thus, each system release consists of a selection of features that customers find valuable. Additionally, every version serves to address flaws found in earlier product variants. Release planning (RP) deals with choices on the features that should be chosen and assigned in order to produce a series of subsequent product releases that adheres to crucial technical, resource, financial, and risk constraints.

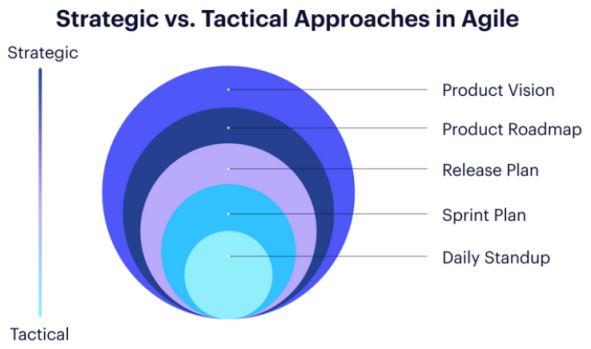

Fig. 1. Strategic vs Tactical approaches in Agile [9]

While a release plan does not attempt to plan for years in advance, it does explain upcoming releases, and because it is more detailed than a product roadmap (high-level scope and timetable), it differs from that document. An Agile release plan, however, does not list the tasks for each release. Instead, it compiles several sprints or iterations into releases. In actuality, a release plan guarantees that the team consistently produces a coherent version of the product. It's a tool for integrating adjustments that will have a big influence on the user experience in a short amount of time.

Including numerous releases and maybe even multiple projects, road maps provide a longer-term perspective. They seek to capture the product vision since they are strategic instruments. They present high-level features and product capabilities as well as product and release goals.

Release plans are more detailed and have a shorter time horizon. Compared to roadmaps, they are more tactical, focusing on the precise work that has to be done and providing information down to the level of individual backlog items.

One large software project is divided into numerous executable sub-projects during agile development. They start by completing the most crucial user-selected functionalities and then employ iterative incremental development techniques to create working systems after each cycle. The development team concentrates on responding rapidly to the shifting needs. The development team promptly modifies the plan when demand changes. Once proposed, it is well received by small and medium-sized businesses due to its quick response to changes in user demand.

B. A good release plan should

- provide maximum business value by offering the best possible blend of features in the right sequence of releases
- satisfy the most important stakeholders involved
- be feasible with available resources, and
- reflect existing dependencies between features.

III. The Art of Release Planning

The implicit and tacit parts of RP are addressed by this method. Since RP is a problem with a vague definition, it takes human intuition and capacities to describe the issue before looking for a solution. An organization's lack of focus on processes in general, whether unintentionally or because of a lack of RP process maturity, can be one of the reasons for approaching RP as an art.

This more humanistic approach is also used in agile development, which makes use of the well-known benefits of short, iterative software releases to get early user feedback. In agile development, RP focuses on planning for the following iteration. For this planning process to work, the key stakeholders must physically/virtually meet in order to discuss and explicitly negotiate which features to create next and how much work they would require.

When several stakeholders are involved, RP in agile development doesn't offer instructions on how to decide on features and priorities8. Additionally, it doesn't offer any methods for balancing competing demands from various stakeholders.

Even in more plan-driven workplaces, RP is frequently carried out fairly informally and relies on simple tools like spreadsheets. The key stakeholders are manually contacted and their interests and preferences are weighed against the resources at hand when making release decisions. However, as the number of features and stakeholders increases, the exponential growth in the number of potential release plans outperforms the power of manual plan generation.

IV. Release Planning Process Activities

Release planning should happen when a team is ready to begin working on a new release (this could be after a previous one is complete or when starting

on a new product). This would occur, however, after the roadmap has already been planned and confirmed [1].

This involves the following: establishing clear, specific, and measurable goals. that describe the outcomes or benefits your product should create. The ”what” questions of the project, such as ”What will be done, what can we give as our end product?” are answered through release planning.

- Through close collaboration between the development team and the customer, each feature of the product is dissected into a user story
- The technical team members, the product owner, the scrum master, and the other members of the development team all participate in these meetings
- Measurable goals that specify the results of our product should be established during RP
- The development team then chooses the user stories for the release and calculates their length
- In light of the fact that they can identify the release features, the product owner then prioritizes these tales based on their importance
- According to their estimates of the user stories, the product manager and the development team estimate the release date, and the product owner obtains the development team's commitment to the delivery date
- A product owner can be in charge of releasing goals, writing epics and user stories, and initiating the acceptance criteria
- On the other hand, the Scrum master will organize and moderate all meetings, run regular reporting, or simply consult with other team members for best Scrum practices

The specific terminology might vary a bit among Agile practitioners. The basic framework spans the product vision, product roadmap, release plan, sprint plan, and daily standup. Release plans are shorter-term and decidedly more granular. Release planning should include making data-driven decisions about which user stories to use based on actionable feedback from team members.

V. Iteration Planning Process Activities

In Iteration Planning (IP), the product owner and the development team work to establish a common iteration goal. These iterations, sometimes called sprints, occur in an operational environment, where software teams test business and technical solutions while simultaneously delivering the product. The work of the team members is to estimate the work that will be completed in each iteration.

Iteration Planning essentially works to Answer the “how” question (i.e., “How are we going to get there?”) This method supports the team in executing their goals quickly, getting feedback, assessing performance, and improving their processes [1].

- During this planning meeting, the team organizes and self-assigns the upcoming tasks. The product owner facilitates the discussion here as well and the Planning process ends once the team agrees on the goal
- The team quantifies their capacity to perform work in the upcoming iteration. Each team member determines their availability, acknowledging time off and other potential duties
- Each user story should support the iteration goal and so, the Development team, which usually includes the technical members, chooses the particular user stories stored in the product backlog
- If the team, however, finds any of the user stories that do not fit in the current iteration, they can remove them and this decision is taken with the help of the product owners and technical teams
- The development team creates the backlog tasks that work with each user story. Team members break the user stories into individual tasks and allocate the hours taken to complete them
- If a task exceeds the hours available, they will discuss with the product owner what tasks/user stories are the best to remove. If extra time is available within the iteration, they can include another user story

In IP for a large project, an up-to-date product plan is designed with inputs from all the team members. A good understanding of the current state of the project is established which will facilitate the further development of the modules and the proper allocation of work to the members. Immediate objectives are derived from the backlogs of the modules that will be periodically monitored to prevent further failures. Cost estimations for backlogged items are computed which will affect the total estimation of the project cost.

VI. Constraints in release planning

The main types of constraints in release planning are hard and soft constraints.

A. Hard Constraints

These constraints are things that 'Must' be completed before the next task can start. They are often stated to be 'Mandatory' constraints, as in, a prototype cannot be tested until it is fully constructed.

This includes technical, budget, cost, resource, effort, and time constraints in release planning. The time constraint refers to the project's schedule for completion, including the deadlines for each phase of the project, as well as the date for the rollout of the final deliverable. The cost of the project often dubbed the project's budget, comprises all of the financial resources needed to complete the project on time, in its predetermined scope.

B. Soft Constraints

This is of the format,

- Management approval is needed
- Stakeholder approval is required before continuing
- Senior management approval is required before proceeding

Soft constraints 'should/could' be met. They are sometimes called 'Discretionary' constraints. While these constraints are often mistaken for 'hard constraints', they are optional. This includes the influence of stakeholders, value, risk, and resource consumption factors.

C. Challenges in agile release planning

One of the common challenges in dealing with features is that the team should possess a deep technical understanding of desired features that will aid the product is required [2]. Another challenge is balancing constraints with functionality.

With Agile specifically, strategies and goals may change between each iteration and it can never be permanently fixed. These changes are hard to deal with since certain functionalities might have to be altered on a larger scale with each iteration. This increases uncertainty in project velocity, cost, and business value.

VII. Approaches to Release Planning

There are a number of commonly-used methods to assist with the planning process in agile project management. Here, we will discuss two such approaches.

A. Planning Poker

The first approach is referred to as "planning poker" [3]. Planning poker is a method for reaching a group consensus on estimated time and effort required by a project, based on common sense and experience. It is a simple and collaborative method, and can be used to estimate any number of metrics, without modification to the process.

The steps required are as follows:

1. Each member of the development team is given a set of cards, each of which is labeled with one in a range of numbers. The group will agree beforehand upon a scale for estimating a given metric. Typically, this is either number of story points or ideal days, and a higher value indicates greater expected effort.
2. The moderator reads a user story to the group.
3. The group discusses the story to resolve any confusion and share insight.
4. Each participant casts their vote by holding up a representing an estimate.
5. If there is high discrepancy among the votes, then the participants with the highest and lowest values will share their thought processes with the group to bring the group closer to a shared understanding. Then, the moderator holds another vote.

Step 5 repeats until votes converge on a single, agreed-upon estimate.

B. Monte Carlo Methods

The second category of estimation approaches that we'll discuss are Monte Carlo methods. "Monte Carlo" methods refer to a class of strategies that are used to make decisions in the presence of uncertainty. In the context of Agile release planning, this typically relates to estimates of story size, business value and project velocity (i.e., how much work can be completed in a single iteration) [5].

A Monte Carlo approach generally consists of the following steps:

1. Define a range of possible values with three values: a pessimistic estimate, an optimistic estimate, and a most-likely estimate.
2. Let o, p, and m represent the optimistic, pessimistic, and most-likely estimates, respectively. Use these three parameters to define a distribution. There are two commonly-used distributions: the Triangular

distribution, and the Project Evaluation Reviewed Technique (PERT) distribution, discussed below.

3. Repeatedly sample random values from the above distribution. An appropriate number of trials is typically in the hundreds or even thousands. Record each sampled value, and when the simulation is complete, the histogram of generated values can be used to determine the likelihood of values that may be observed during the project.

The distributions mentioned above can be defined the following way:

Triangular distribution [4]:

$$\sigma = \frac{o^2 + m^2 + p^2 - om - mp - op}{18}$$

$$\mu = \frac{o + m + p}{3}$$

Project Evaluation Review Technique (PERT) distribution [6]:

$$\sigma = \frac{p - o}{6} \quad \text{and} \quad \mu = \frac{o + 4m + p}{6}$$

These differences are illustrated in the figure:

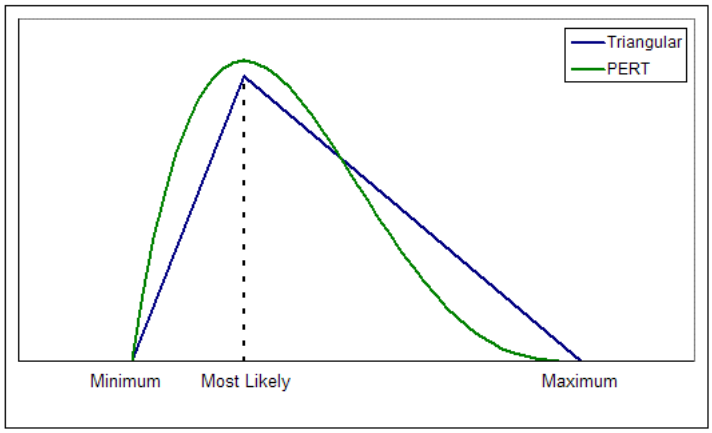

Fig. 2. Triangular distribution vs. a PERT distribution [4]

The choice of distribution for a Monte Carlo simulation is left to the development team. The main difference to note is that the Triangular distribution provides a more "conservative" estimate, having a fatter tail in the skewed direction, whereas the PERT distribution is more heavily-concentrated in the direction of the initial "most likely" estimate. Another difference to note is that the PERT definition results in a smooth, differentiable curve, whereas the Triangular distribution is discontinuous at the most-likely value, where it reaches a peak.

Generally, a point of vulnerability for using a Monte Carlo simulation is the element of human judgment in selecting a distribution and in estimating the three values that define it. However, human judgment is also at the core of Planning Poker. These two approaches exemplify how it is possible to reduce uncertainty in the planning process, but difficult to entirely eliminate it.

VIII. Conclusion

There are a number of factors contributing to the difficulties surrounding Agile release planning. Teams not only need to balance desired functionality against available resources and skills, but they also need to work around the uncertainty that is inherent to Agile contexts.

When contending with the challenges mentioned above, it is important to keep in mind the advantages of Agile software development. These benefits have been thoroughly researched and are well-documented. A 2020 McKinsey [7] study of 22 organizations in six sectors showed that an "Agile transformation" led to improvements in a number of areas.

The study reported the following outcomes of adopting Agile practices:

- 20-30% increased employee engagement
- 10-30% increased customer satisfaction
- 30-50% improved operational performance
- 20-30% improvement in financial performance

Another study by Diebold and Mayer [8] considered the effects of Agile practices – for example, sprints, user stories, and planning poker – on project metrics such as risk management, understandability, and planning. The study found that of ten metrics considered, all of them benefited from at least one agile practice.

Studies such as these further support the idea that Agility is desirable and effective. This is important to remember as we encounter challenges inherent to the planning process, and a reminder that it is worth striving toward innovation in planning practices.

REFERENCES

[1] G. Ruhe and M. O. Saliu, "The art and science of software release planning," in IEEE Software, vol. 22, no. 6, pp. 47-53, Nov.-Dec. 2005, doi: 10.1109/MS.2005.164.

[2] S. Alyahya, M. Alqahtani and M. Maddeh, "Evaluation and improvements for agile planning tools," 2016 IEEE 14th International Conference on Software Engineering Research, Management and Applications (SERA), 2016, pp. 217-224, doi: 10.1109/SERA.2016.7516149.

[3] N. C. Haugen, "An empirical study of using planning poker for user story estimation," AGILE 2006 (AGILE'06), 2006, pp. 9 pp.-34, doi: 10.1109/AGILE.2006.16.

[4] Triangular Distribution. MATLAB and Simulink. https://www.mathworks.com/help/stats/triangular-distribution.html

[5] K. Logue and K. McDaid, "Agile Release Planning: Dealing with Uncertainty in Development Time and Business Value," 15th Annual IEEE International Conference and Workshop on the Engineering of Computer Based Systems (ecbs 2008), 2008, pp. 437-442, doi: 10.1109/ECBS.2008.58.

[6] Tysiak, Wolfgang. (2011). Risk management in projects: The Monte Carlo approach versus PERT. 906-910. 10.1109/IDAACS.2011.6072904.

[7] Aghina, W., Handscomb, C., Ludolph, J., Róna, D., West, D. (2021, March 1). Enterprise agility: Buzz or business impact? https://www.mckinsey.com/capabilities/people-and-organizational-performance/our-insights/enterprise-agility-buzz-or-business-impact

[8] Diebold, P., Mayer, U. (2017). On the Usage and Benefits of Agile Methods Practices. In: Baumeister, H., Lichter, H., Riebisch, M. (eds) Agile Processes in Software Engineering and Extreme Programming. XP 2017. Lecture Notes in Business Information Processing, vol 283. Springer, Cham. https://doi.org/10.1007/978-3-319-57633-6-16

[9] Racasan, M. (2022, November 21). Agile Release Planning: A guide. Zenhub Blog. https://blog.zenhub.com/what-is-a-release-plan-in-agile/