

Demystifying Story Points

Author: Zaheerabbas Shaukatali Contractor, Co-author: Swati Garg

Consultants, Agile Center Of Excellence, Wipro IT Business

Abstract

Software practitioners find it challenging when they are asked to adopt Agile ways of sizing and estimation of software which is yet to be developed using Agile methodologies. At Wipro as we encountered same challenge, Among several approaches that we tried for Agile estimation, we are sharing the one that is well received by practitioners from conventional background. We found remarkable improvement in estimation capability of team as we adopt this approach for a first time agile team. This paper elaborates on lessons learnt through a case-study to develop step by step approach for addressing the Agile estimation and sizing challenge encountered by first time Agile teams.

Introduction

We are part of the Agile Center of Excellence (COE) at Wipro Technologies. The Agile COE consists experienced mentors and coaches who provide extensive coaching and mentoring covering various aspects of Agile to hand hold the project from first time Agile adoption to the cases where an specific business challenge requires attention for resolution of pain areas. The Agile COE also undertakes the study of aligning the governance parameters at the organization level while balancing the Agile Practices.

Background: Software sizing versus Estimation

To quickly recollect, an estimation process consists of taking inputs on project size and transforming it to effort & schedule which is a direct measure of cost. Conventionally the project size has been the ultimate measure of the software in units like lines of code, function points, use cases and so on. These standard measures help for estimation through benchmarking at the organization levels across various projects. Repeated trend of co-relation between project size and efforts help an organization and its projects to establish a pattern of productivity under influence of several variables such as technologies and domain expertise, skills and experience level of the project teams. Availability of pattern of productivity for different cases enables reliable estimation of future projects in terms of expected effort or time to develop, if the size of project is known. Effort is subsequently transformed into number of days to develop depending upon availability of resources and interdependencies of tasks and activities.

Often conventional methods of sizing fall short to include the fluid behavior software complexity and algorithm, yet the big black box approach of execution apparently creates a perception of reliable estimation. This perception is tough to break even though several projects estimated in conventional manner reach the point of re-estimation sooner or later during project execution period.

Traditionally sizing and estimation of a project is performed by few key stakeholders and hence it's seen as their responsibility to make it possible. Team contributes as per directives that are more about effort as per project plan than size of activity to complete the project. As a result the team members cannot comprehend the importance of size until late. They rather sync into an understanding that effort spent on an activity is direct measure of its size.

This insight on project team member's understanding led us to retrofit approach of effort to sizing which ultimately helped them to realize the difference between two and embrace the story point way of sizing for Agile projects.

Story Points

Shift to Agile can be compared to culture shift and with that comes typical culture shock in terms of requirements stacked in the form of Stories and estimation of stories done in Story Points at the Release level (Approximate Estimate) and Sprint level (Concrete Estimate). Story point as the name suggests have been derived from "user stories" which is commonly expressed for requirements in the Agile projects. Story points are relative measurement of the size and complexity of the user stories wherein a base story is assigned some story point/s to start with and rest of the stories are estimated in story points based on its size and complexity compared to the base story.

Adoption Challenge of Story Points

For any new comer to Agile, fitting the stories into time box schedule by vertical slicing them with the story point sizing based estimation is a huge task and often turn out to be a disaster to start with in initial few iterations especially in absence of a shop floor Agile coach or mentor. Situation becomes tricky in for the Software service industry wherein the customer could be the latest entrant in the Agile enthusiast list and asks for the size, cost and fixed schedule for the planned release.

Customer may carry the expectation from the delivery team of being able to absorb requirements volatility in the fixed schedule there by considering Agile as the silver bullet!

Epic Unfolds

In this section we sharing the case study through sequence of episodes that led us to solve the challenge that we rarely realize till we followed conventional approach of estimation.

Before on boarding Agile for current and future releases, this project was executed in waterfall manner and hence we were comfortable in using the Work Breakdown Structure (WBS) approach for estimation.

As customer chose to use Agile, they really wanted the team to adopt as much Agile practices as possible, hence Agile estimation based on Story points was one of the clear expectation. Nevertheless, they also wanted affirmation on effort and schedule for current and future releases.

In first iteration, it was almost not possible to think in terms of Story points and the key project stakeholders decided to continued the WBS way, yet unlike past, under influence of mentor from Agile COE they involved the entire team while doing WBS estimation.

. Project Manager who earlier used to do it all with thoughts from few more leads, were asked to guide the team to decompose the user stories into detailed tasks by going to the lowest level using the Work Breakdown Structure (WBS) technique. The project team came up with their efforts for detailed requirement elicitation, design and CUT and testing activities at the story level..

The idea was well received and with some amount of mentoring and guidance the team members came up with the efforts for implementing the story with the doneness criteria. Interestingly as we completed this exercise, project team unanimously asserted that it was a vital change and they felt more

committed to estimates made by them. It boosted our confidence in approach and at the same time Project Manager shared that he could sense a new wave of energy in team that he was looking for.

The effort estimated for the stories by team was actually the Ideal Engineering Hours to completely implement the respective story because no interruptions were considered during the WBS estimation for each story.

Based on prevalent studies Agile COE recommended the team to consider the Ideal Engineering Hours of respective story as its Story Points and the team followed the suggested approach for subsequent iterations where in they would do the WBS to churn out the Ideal Engineering Hours (IEH) and then equate some IEH to one story point and define the commitment level for iteration contents. For example: Team can choose to consider 5 or 10 IEH equal to one story point.

However, soon we realized that in a typical outsourcing scenario the teams are distributed amongst various vendors and customers. This creates a situation where the distributed team comes with a varied experience in terms of domain and technical knowledge apart from the various other factors like infrastructure availability from the execution perspective. This creates a situation where different distributed team will come up with varied efforts based on the above mentioned factors. For same amount of work an expert will take less effort to develop as compared to a rookie. Equating Ideal Engineering Hours to story points in such a case would mean that expert would have less story points done than a rookie. It is a false negative notion. Also at the end of iteration if both of them have worked for same amount of Ideal Engineering Hours than there is no difference in their respective productivity, which is again a false negative notion.

Lessons Learnt

Our failures with story points sizing in the case-study above, brought in an important aspect of Velocity and the Story points. Each individual's measurement for story points of a particular story should not be different based on one's capability. However, each individual can take different efforts and the timelines to complete a story. This pattern could be gauged over the period of time based to assess individual's productivity—which is essentially known as velocity of individual based on various factors.

Let's take an example: Time taken to climb up from ground level to 5th level, should not be a direct measure for height climbed, because a lift could take much less time than an ordinary adult. A small child may take more time than an ordinary adult.

Essentially the height to be travelled should be measured as number of steps between respective floors and total number of floors to be climbed. It is also important to consider the direction of travel because effort and time required to climb up 5 floors from ground level is not same as stepping down from level 5 to ground floor.

Based on learning from the case study as above, through this paper we are sharing what we now suggest to our teams who were previously following WBs kind of estimation:

1. Do not restrict the estimation task to key stakeholders. Make sure that the whole team participates in estimation from beginning.
2. Guide the team members to lay down their respective WBS for the stories that are chosen to be implemented in the sprint/iteration.
3. Do not equate Ideal Engineering Hours to Story points, because it will vary based on individual's expertise.
4. Guide the team on how to identify the key functional aspects of story which are going to be same for everyone.

Guide the team to identify the smallest functional component as one or more story points.

5. Each story should be measured with respect to the smallest components that are allotted definite story point.
6. Promote practice of the steps 4 and 5 to ensure that all team member measure same or closer story points for respective story.
7. Keep the team informed that different team members may take different time and effort to develop same amount of story points depending upon their initial velocity. However, in future iteration, as part of team they should target to attend the velocity benchmarks established by experts and mentors in the team.
8. Experts and mentors should be accessible to team till the moment team becomes comfortable in unified Story Points Measurement and collectively strives to touch the benchmark of Velocity and reach beyond it.

Conclusion

Story point sizing based estimate is not limited by external laws of functionality measurement unlike conventional methods. A team can define its own rules of measurement of functionality to encompass different real time aspects such as familiarity with technology, complexity of implementation, experience in domain and more. In such case we get constant measurement of story points within one project context. We learnt that time taken to climb cannot be a direct measure of height climbed, in other words we do not support the approach to consider the effort/Ideal Engineering Hours to be used for measurement of Story Points.

Reference

- User Stories Applied: For Agile Software Development By Mike Cohn ISBN: 0-321-20568-5
- Agile Estimating and Planning By Mike Cohn ISBN: 0131479415, 9780131479418