BelieveIT
We Deliver ....

Agile Project Tracking

Introduction

The following white paper assumes that a plan has been created based on the "Agile Project Planning" white paper available on BelieveIT 's ProcessMashup website.

This document describes key principles of a Progress Tracking model : Its not about "a task should be completed on a specific day", its all about, how much progress has been made against the tasks that need to be delivered.

This approach allows the developers to pick up the tasks they have to complete in any order they want; Progress can be measured accurately as we track the amount of work that needs to be completed against the amount of work that has been completed so far.

So lets get started ....

WHY TRACK ?

- KEEP PROJECT UNDER CONTROL
- FIND PROBLEMS EARLY
- DELIVER UNDER BUDGET
- MEET PROJECT COMMITMENTS

See also, Sample MS Project Plan, found at http://www.processmashup.com

Preparing The Plan

We have the project plan for the project that has been running for a week—now we need to update the business on progress.

To begin with, we need a couple more columns to be in view in the plan : "Actual Work" and "Remaining Duration". On the assumption that we already have the columns "Work", "Duration" and "Start" and "Finish" Dates.

Actual Work : Contains the number of Man days or hours completed against the task, this is used to calculate the progress made against the project delivery.

Remaining Duration : the number of real days to complete the remaining work on the task : This is the report we need from the developers.

]

Task Name	Remaining Duration	Actual Work	Work	Duration	Start	Finish
Map the customers requirements into User Stories	**22.5 days**	**0 days**	**12 days**	**22.5 days**	**Mon 07/03/11**	**Wed 06/04/11**
Analyse Requirements	6.25 days	0 days	5 days	6.25 days	Mon 07/03/11	Tue 15/03/11
Create Stories	6.25 days	0 days	5 days	6.25 days	Tue 15/03/11	Wed 23/03/11
Add Functional Requirements	10 days	0 days	2 days	10 days	Wed 23/03/11	Wed 06/04/11
Review with Team	3.75 days	0 days	3 days	3.75 days	Wed 30/03/11	Tue 05/04/11
Understand how the requirements will be implemented	2.5 days	0 days	4 days	2.5 days	Wed 06/04/11	Fri 08/04/11

Getting the Progress Updates

To track progress you need a report from the delivery teams, stating; what tasks have been progressed, which have completed and how many elapsed days are needed to complete the tasks that were progressed, the team should also let you know of any problems that may be upcoming or problems that are impeding progress. How you get this information will depend on the team structure, meetings are preferred but may not always be feasible, an email report from the Scrum Master / Tech lead would do.

The next section will walk through updating the plan.

Actual Progress

The reports from the team tell us the elapsed time needed to complete the tasks progressed; Update the plan for each of the reported tasks i.e. adjust the 'actual work' field of each task, so that the remaining duration of each task equals the amount of time reported by the team needed to complete the particular task.

At the start of projects I would normally understate the progress to give the team a bit of padding as it is almost certain that something unexpected will crop up. So, if I am told; 2 days remain, I will leave between 2 and 3; 5 days remaining, I would leave about 6.

The logic being, if the estimate is correct then there will be a positive jump in delivery next week, if I am right then progress will be as expected next week, if the task slips then it will be less of a slip.

Once all the tasks reported have been updated, save the plan remembering to take a note of the summary value of the 'Actual Work' field. In this case (the plan on the previous page has now been updated, see below) that's 6 days, this means the team have completed 6 man-days work on the project.

Task Name	Remaining Duration	Actual Work	Work	Duration	Start	Finish
Map the customers requirements into User Stories	**15 days**	**6 days**	**12 days**	**22.5 days**	**Mon 07/03/11**	**Wed 06/04/11**
Analyse Requirements	2.5 days	3 days	5 days	6.25 days	Mon 07/03/11	Tue 15/03/11
Create Stories	2.5 days	3 days	5 days	6.25 days	Tue 15/03/11	Wed 23/03/11
Add Functional Requirements	10 days	0 days	2 days	10 days	Wed 23/03/11	Wed 06/04/11
Review with Team	3.75 days	0 days	3 days	3.75 days	Wed 30/03/11	Tue 05/04/11
Understand how the requirements will be implemented	1.5 days	1.6 days	4 days	2.5 days	Wed 06/04/11	Fri 08/04/11

Planned Progress

We are now going to calculate the work days that should have been completed in order for the project to deliver as planned.

Now, save the plan to a new name, I normally name the plan with the date, and suffix it with "planned". Now go to the top of your plan, to the summary task in the "Actual Work" column and set the value to 0. Now check to ensure that "Actual Work" has been zeroed in any of the sub cells of the tasks that we want to check.

Now, go to the Project Tab in MS Project and select Update Plan, select "update work as complete through 0-100%" and set the date to Yesterday as this was the day the report was created and todays work has not completed . Select "Entire Project", and Click Ok.

Read the value from the summary field on the Actual Work Column, this is now showing the amount of work that should have been completed until close of business yesterday. The value in the example is 10.5 days, subtract this from Actual work total in the plan we updated and we are 4.5 days man-days behind schedule, so we have a problem to resolve.

See the plan below.

Task Name	Remaining Duration	Actual Work	Work	Duration	Start	Finish
Map the customers requirements into User Stories	**7.5 days**	**10.5 days**	**12 days**	**22.5 days**	**Mon 07/03/11**	**Wed 06/04/11**
Analyse Requirements	0 days	5 days	5 days	6.25 days	Mon 07/03/11	Tue 15/03/11
Create Stories	0 days	5 days	5 days	6.25 days	Tue 15/03/11	Wed 23/03/11
Add Functional Requirements	7.5 days	0.5 days	2 days	10 days	Wed 23/03/11	Wed 06/04/11
Review with Team	3.75 days	0 days	3 days	3.75 days	Wed 30/03/11	Tue 05/04/11
Understand how the requirements will be implemented	1.5 days	1.6 days	4 days	2.5 days	Wed 06/04/11	Fri 08/04/11

Reporting

So the project is a bit behind schedule, but we have 3 people on the project so we are only 1.5 elapsed days behind plan, but this is in the first week which is not good, so what can we do ?

Look at the two plans, where is the difference in performance? Which tasks are not progressing as fast as planned? Talk to the relevant developer and find out what the problem is.

So you need to report that progress is not as anticipated, but provide an explanation based on the result of your discussions. If the slip repeats in the next report then a rethink on the current plan is needed.

Success Criteria

- This Method only works if the plan is complete and is tracking all activities required to complete the delivery. If items are missing, then the calculations will be wrong. Do keep the tasks to a maximum of 5 days and a minimum of 2 days duration, you don't need more information than this.

  For example :

Task Name	Duration	Work
Get Catalogue Data	**22.5 days?**	**18 days**
UI	**22.5 days?**	**12 days**
Retrieve Search Text	3.75 days?	3 days
Display Results	3.75 days?	3 days
Process Selection	3.75 days?	3 days
Test Story	3.75 days?	3 days
Backend	**5 days?**	**4 days**
Add search to Stock Service	2.5 days?	2 days
Check API Compatibility	2.5 days?	2 days
Unit Test	2.5 days?	2 days

- It is important that you roll up tasks and don't break them down to far! The completion of the task needs to be measurable, so complete design of story US12345, or, complete UI for story US-1233, if you break down the tasks too far the plan becomes unmanageable and unreliable.
- The team created the estimates and tasks and have committed to delivering them, so work with the team. When you put the plan together you got your change to put in padding in case of problems. So, if things start slipping it is a joint responsibility and not the delivery teams fault.
- Don't apply too much pressure on the team, you need them to be open and honest with you, if you brow beat them when they report a slip then quite often they will try to cover up the slip and the project will end up slipping and you won't know about it.
- Talk to the team members, if you can, to get the updates - I wouldn't rely on using one of the tracking tools to get an accurate view of project status as they tend not to be kept up to date.

Conclusion

I have found this model very useful over the years and haven't had a project slip for at least 8 years. Things do go wrong and occasionally they need to be re- planned, but this is the PM's job.

Sometimes you get pushback that "this is not Agile" and "the point is Agile is not planned", well to be frank if you want to deliver software projects, the team need to understand what has to be done and this process ensures they do. I regularly go into large organisations where the teams' last Agile project slipped massively, perhaps to 2 Years! I apply these simple practices, and the same teams deliver to plan, quite often under budget. This is what every business wants and needs; product delivered fast and reliably.