

Student Name: Bill Carswell
Homework Assignment: #9
PMGT501
Date: March 1, 2010
ePortfolio Matrix Correlation: Planning: Time

In 300+ words, outline a schedule development process for small projects

In the small project environment the schedule is the easiest tool for the project manager to use for project execution, monitoring and control. With resource allocations made it is a convenient action tracker and with every deliverable item accounted for there is no need for separate earned value management, as long as the customer does not require it.

There are three keys to an effective schedule: Keep it simple, build it around a clearly defined and comprehensive list of contract deliverables, and track all risk reduction activities.

In order to keep it simple, capture key milestones and deliverables and the top-level efforts required to meet those marks, but avoid adding too much detail for those efforts. The team typically knows what they have to do to get their jobs done so don't clutter up the schedule with common sense details that will get done whether you track them or not. If you hold your people accountable for their work and update their status regularly, anywhere from daily to weekly, the details will get done. It's also important to capture activities that get mentioned in meetings and discussions that aren't so obvious in day-to-day work. All too often someone will throw out a good idea during a meeting that everyone agrees needs to be done, but it doesn't get written down and it gets forgotten until it's too late. Capture those ideas in the schedule and assign them to someone in your resource pool.

Effective schedules are also built around deliverables. After all, the deliverables are the reason for the work. So every activity should lead to a deliverable, and every project deliverable should have its own line item somewhere that shows it being handed over to the customer. In good project management practice these deliverables are all accounted for in the Work Breakdown Structure, which is created before the schedule is built. Do not build a schedule in your favorite software tool, then hit the "create WBS" button and number your schedule. This is not a WBS, it is just a numbered schedule. Build the WBS, then build the schedule and make sure it accounts for every WBS item. Also be careful not to include any schedule item that doesn't contribute to a deliverable. If it doesn't contribute to putting a contract deliverable in the customer's hands it is out of scope and doesn't belong in your project schedule.

Finally, make sure you integrate all your risk mitigation activities into the schedule. These are the most commonly overlooked activities on a project, but often some of the most crucial. It is easy to build a basic schedule, then conduct a risk mitigation exercise and diligently identify risks and associated mitigation efforts. But if those risk mitigation efforts don't get captured in the schedule they often don't get done because they aren't

obvious day-to-day and the project team is usually so busy executing the primary plan that they don't think about the risk efforts. Capture risk activities in the schedule and status them just like you status all the other schedule activities, and your risk management is much more likely to be successful.

My recommendations for developing a schedule are:

1. Start by capturing major milestones, like start dates, contract completion dates, reviews, tests, training, demonstrations and other known events.
2. Then consider when each deliverable (WBS item) needs to be completed your contract obligations and capture those as deliverable milestones, starting at level 1 and working your way down to lower levels. This helps paint the big picture and starts establishing the project's schedule logic. Make sure every single WBS item is in the schedule and has a delivery date, and identify the resource (person) responsible for the item. This is critical.
3. Capture the main activities required to produce the WBS deliverables. Work with the responsible parties for each WBS item. You might want to start with only 3-4 activities for each WBS item. Again, keep it simple. Sometimes you will need to add more, but don't get too bogged down in details to start with. A simple schedule can be more effective than one that is too detailed, because overly detailed schedules are difficult to see and often get ignored. Make sure people are assigned to each activity
4. Consider any long-lead items that may require special attention. For example, if you are building a surveillance system and need to buy radars that have a 9-month lead time and those radars need to be installed and operational in 12 months, you might consider putting their procurement on the schedule. On the other hand, the procurement of nuts and bolts that are easily obtained don't need to be tracked on the schedule. Your team members will make sure that gets done.
5. Visit your risk register and make sure every risk mitigation activity is captured in your schedule. It is also convenient to make sure the schedule is organized so that the risk mitigation activities are associated with their related deliverable items. Again, make sure someone gets assigned to each activity.
6. Realize that the schedule is a living document. Of course, you have to have a "baseline" to establish your plan, but details can be added as long as they don't affect milestone dates. As new activities arise from ongoing activities and as new risks are realized, capture new key activities.

This type of schedule is an easy project management tool. It provides you with a quick sense of the status of deliverables and milestones. You can collapse efforts associated with each WBS item so that only the WBS summary lines show and use it in team meetings to provide the team with a quick snapshot of the big picture. And since all the activities are resourced, it's easy to filter on a particular individual and create a personalized action list for a each individual that you can review in private one-on-one meetings.

It should be noted that this essay applies to small projects that typically engage a small core team where everyone knows each other and span from a few weeks to a year or so.

I'm also assuming there are no human resource issues, the team is assigned and relatively unchanging, and there are no project manager responsibilities for professional development of the staff. The project is a hardware or software deliverable, although it can be a research and development effort.