The demented rambles of a project manager,

By David O Neill

Introspectively analyzing myself and of the other team member's, the dynamics of the team with varying contrasts and backgrounds, it's hard to see the value of the outcome of the project set forth. Without real life input into such a project plan, only a hypothetical result could ever some from this.

However as many people would probably see this as a deterrent, striking off such an undertaking of such a hypothetical exercise, I believe what is really the learning outcome of the project is the process.

Given this strong opening statement it was clear to see straight off the problems our team would be faced with. Right at number one of organization of schedules and times. With varying backgrounds and academic differences bringing a team together was a difficult process, with three weeks going by before we actually met as a group.

I would imagine that if we were to draw the triangle, scope, time and money that uniform equilateral triangle had already morphed into a scalene triangle; with time demising, the cost of this will soon be realized.

As we met after week one, the dynamics of the team seemed good, most team members had input, but scoping the exercise was difficult as the determination of what exactly was to be produced by the company was infinitesimal un-deterministic. The only information given to us was that the new branch of supermegaglobalcorp was to produce an internal product within one year of setup.

An internal product, three of the team members had previously done two modules in object oriented analysis and design and system architectures. Having this information under their belts proved somewhat disruptive and deciding on what exactly was being built would prove difficult. Two of the other team members where not familiar with software processes and the idea of creating bespoke software in superglobalmegacorp did not entail recreating the wheel, but rather extending the wheel with some functionality.

As with most corporations' internal bespoke software solutions have for the most part already been established. A new product would either replace an old product, which would probably be dependent on some other product. For example changing out AIB's websites but the database of information staying the same, or extending or creating a new interface to an antiquated system.

From an engineering perspective this would be hard to grasp, the idea of taking part of a building and using it to build another is probably not the best plan, but in most software solutions this is almost always the case.

After the struggle with this realization, came the complexity of organizing some sort of schedule. Creating a software product within a year of opening in a global corporation is probably not realistic. A more wise decision of creating a new branch would be integrating this branch into current operations slowly moving them way into their own projects.

From personal experience, I can recall when the Team lead of the Windows Servers Services Team in EUEC Intel announced he was changing position. I can recall the next in line saying that it would take him a year to train in someone new regardless of the background experience, as with large corporations, its not about what you've learned from a book or even the processes of another company. This will not prepare you for the process structures and policies that will take time to learn coupled with the fact that the person training you in will also have their own work to do.

This brought about the discussion of the workforce required and the resources needed to make this happen, after a bit of discussion we agreed that the parent company would have to lease us out already seasoned developers who are familiar with the software architectures already present within the company, the policies, guidelines and tools to be used internally.

After this discussion we then finished up our somewhat abstract overview of the project and the difficulties that we presenting themselves we documented. We then broke the meeting agreeing on work to be done by each individual to be presented for the next meeting!

As the work broke down and the vision of what had to be achieved was realized. The proposed one-year dead line, a constraint within the project definition probably occurs over and over again in project management. The OASIG Study (1995), *"The IT project success rate quoted revolves around 20-30% based on its most optimistic interviews. Bottom line, at best, 7 out of 10 IT projects "fail" in some respect"*.

This brings about the question of why are they failing, I would hazard a guess that inability to scope a project, poor analysis and too many technical, people, and time constraints are present coupled with business overzealous goals to push a product out the door within a set time frame. Many people have said that programming is a creative process; do artists put a time frame on their masterpieces? Given this statement, I would not fancy the job of the project manager in this project, as I believe we are immediately met with an unattainable goal. A sane project manager would probably put a buffer on a buffer, leading to 3 years project time.

As a project manager in this situation its understandable the difficult position that project managers are cast into. Beginning with requirements analysis the process of defining a project in order to meet the conditions and or capabilities defined in a contract. The project assigned had very little in requirements specification, it was very difficult to determine the scope and as a result determining the other parts of the project management would be difficult. Beginning week 3 on the topic of monitoring and controlling, the determination on whether the project was on course and within constraints, the scrum method would be best suited in determining this. From the construction point of view the end goal and resources required are quantifiable, however in this project it was extremely difficult to determine the amount of work needed and the determination of the time allotted to sub tasks.

Therefore settings timelines as seen in the gannt in the project are almost certainly incorrect and would be determined during the process and should be updated accordingly. However the baseline given could be used to determine a more realistic timeline in conjunction with the work completed. Micro management from an early project perspective may be advantageous in getting the project on a given path moving to a more lax approach later in the project. This micro management would give the project manager the ability early on to determine the risks and if work should be dropped. This should help to reduce overall changes required at later stages in the development, of course a skilled project manager would be required to identify these potential risks.

As the project manger has committed to a timeline it could be detrimental to be increasing the time late on, from an investor point of view. If they realize ¾ of the way into a project that its actually going to take another 6 months, planning and launch dates could affect business that depends on the work being completed. And as such lead to a rush job at the end of project and reflect poorly on the project manager.

The influencing factors of the client, stakeholders and business analysts in terms of change control should be managed during scope determination, and any subsequent changes be understood to affect the timeline of any such negotiation. Therefore goals and expectations should be reduced early on to facilitate change when it is determined that a particular change is essential for a successful product deployment.

Project scope management in week 4 we looked at determining scope as part of the project process. I believe this is a terrible way of beginning a project. In terms of architecture, I believe that this should be determined well in advance of a project. A company looking for product should invest in a consultant in determining a projects scope and requirements before going to a software house. This would be the beneficial element of having a 3rd party advisement to both the software house and the investors leading to a more realistic

perception of scope, time, money and quality from the parties. This would also mitigate the risk from a project managers point of view as it puts the risk on the consultant who would be in charge of negotiating the parties into a conducive expectations and outcomes.

Collecting the requirements pushed to the consultant would allow the project manager to get on with the job of producing what is required, comparable to an architect been given architectural drawings.   Therefore successful consultants of clients and software houses would wreak their own rewards. Having planning as part of a project can be very detrimental, planning should be before a project with the allowance for change being factored into this plan.

Entering week 4 and the understanding of scope once again the integration of the stakeholders into a software houses concerns could be deemed micro management from an investor point of view. Their invested interests and lack of understanding of what is being produced, has the potential to the disruptive to the process of creating software. The continual change of requirements and expectations from the investor is just going to delay the project. From the point of view of the customer working with the developers should be done at a prototype delivery stage, in conjunction with the consultant whose responsibility was to make sure that the expectations were realized from the customer during the scope and requirement spanning stage.

This would mitigate the need for change management flexibility process, not however negating it, as change is inevitable.   I well defined change management process is required and the impact of a change should be explained to the customer via the consultant. Allocation of resources can be properly determined interval stages, reducing ballooning and inevitably leading to a more defined timely process.

Defining the sequence of activities given the requirements from the consultant, given the work breakdown structure, the decomposition of tasks and effective early timeline perspectives would be used as a way of bidding on a project. This would provide better expectations to the customer, and a realization of exactly what is being delivered and understood. Are 80% of projects considered a failure because the software is at fault or because the customer didn't know what they want?

Mandatory hard logic is the responsibility of the customer to determine and any soft logic should be part of the developing process but should not affect the hard logic.

Critical path analysis is a useful practice in determining time of a project, but many paths can lead to confusion. Parallel development in project can increase confusion and integrating these paths can lad to difficulty, paths should be reduced to a minimum, making sure that each part does what is required before

moving to the next. If the competing paths of development end up with problems with integrating, it's going to increase the amount of work to make them integrate.

Therefore reducing the paths and parallel development should be the goal, integrating forwards and backwards the components. The use of test harnesses to prove the effectiveness of the components, boundary value analysis and the testing of business logic will ensure the requirements of each artifact.
Rather than banging out three components and then trying to integrate them all with one big test harness, or smoke test. Few people can determine chaos, so why emulate it.

Week 5 centered on the idea of budget and cost management. Defining cost in our group project showed the problems with determining cost from a project manager's perspective. Determining salaries, software / hard ware costs and services such as electricity is hard if experience in the area is limited.

For example in the project a team member speculated the cost of 4.6 million for the 2 year project, however, a data center to support 200 developers, with uninterruptable power supply and 'filers' also known as enterprise network attached storages, backup and offsite storage, with fiber switches and fire suppressions systems in a typical datacenter of this size; would be in the region of 5 – 7 million. How could a project manger speculate on these costs without having experience in the area?

As the project manager the cost of these elements should be delegated to experience personal in the area to come up with the cost. Risk is an element to the process of the project, but if the data center burns down would this legitimize the up front expenditure of a fire suppression system. This example shoes the problem with costs estimates and a project manager should be very wary of attempting to estimate an area of which he/she attempts to determine and consult on the area with experiences personnel.

The next three weeks focused on communication and human resource management. Effective and constant communication is key to any relationship, both personal and in business. Without constant communication between the stakeholders and the staff expectations and confidence suffer, therefore any form of communications, whether in the form of status updates, progress reports or forecasts should be frequent and consistent. This skill takes years of practice for the un-accustomed and therefore an effective communications strategy should be developed in order to reduce the risk associated with the lack of communication during the project development.