Estimations in Project Management

Lev Virine, Ph.D.,
lvirine@projectdecisions.org
Michael Trumper,
mtrumper@projectdecisions.org
ProjectDecisions.org

Abstract

In essence, estimates are forecasts of the future, unfortunately, people are not very good at forecasting. While it is difficult to make forecasts of natural phenomena such as the weather; it is even harder to make forecasts of any processes that include people, their knowledge and behavior. Project management is one of these processes.

Estimation is a very important step in modeling and decision analysis. Without proper assessments of project duration, finish time, cost, resources, success rate and other parameters, it is almost impossible to select a proper alternative and ultimately make a good decision.

Our Estimations and What Happen Next

But how do people make estimations? Why do all the activities, in projects or in our lives, take much longer then we originally estimated? Let's analyze a simple hypothetical example that shows how we estimation is done, in this case a software development project. Let's assume that a programmer is already familiar with the scope of the task, but has not done any estimates yet. Here is a conversation between the programmer and project manager at the launch meeting where they discusses a project activity:

"How long is it going to take to build this puppy" ,the project manager asks the programmer about the task.

"Abo-o-out … five days", the programmer answers almost with a slight hesitation.

"Are you sure?"

"Yes, if everything goes well, it should take about five days"

A few hours later the project manager enters this information into the project schedule. As schedules go, it is a masterpiece. It has at least hundred of different tasks and a dozen milestones. Information about cost and resources is well defined and contingency time is added as needed. Even holidays, company meetings, and vacations are entered in to the scheduling software. The project manager prints the schedule and sticks it to the wall. He is enjoying his creation the same way artists might enjoy their new pictures. "We had a big rush and the end of our last project," he thinks, "this time everything should be great."

Now the project starts. Actually, it does not start on time because three programmers are busy cleaning up some problems from a previous project. Then there was a change in the requirements. When it was time to start the task discussed by project manager and the programmer, the project is already late by at least three days. The project schedule is in tatters ignored, everyone knows there is more realism in a Star Trek movie than the Gantt chart hanging on the wall.

Finally, after a delay of three days, the programmer is ready to work on his task. He recalls his original estimate. "I should be able to do it in five days, maybe even faster," he thinks. However, on Monday, he does start coding as he tries to understand the scope of the problem. By the end of the day, he realizes that he doesn't have a clue how to start even though originally everything looked very straightforward. Tuesday, he spends the entire day learning about the new programming tools that he must use. Wednesday starts off with a fire drill and he spends an hour in the parking lot. After that, he spends a few hours on the phone answering questions regarding a previous project. Still he manages to begin coding. At this point, her realizes that that the volume of work required is much larger than he had anticipated. Throw in donut day on Friday, which is slow at the best of times, and our programmer has already burned through his original estimate.

The following Monday the programmer talks to the project manager.

"Everything is almost ready. I had some delays as I had to learn a new tool, but now all of the problems are resolved," the programmer reports during the meeting. ("Almost" means about half of the work in IT language. The project manager knows about it, but prefers to ignore it; he doesn't have a choice). After resolving few major issues and working on some issues that were discovered the previous week, the programmer finally completes the task on Friday .One week longer than the original estimate. The good news for project manager is that this task is not on the critical path. The bad news is that other tasks on the critical path have suffered the same fate..

At this point, the project is delayed by at least one month. Contingency time has evaporated even after dropping few items from the list of features, it will still be difficult to release the product on time. Fortunately, the project manager knows where he can find more contingency time: testing and documentation. Testing is compressed to three days instead of three weeks, who needs testing when you have expert developers and documentation, well nobody reads it anyway.

Software projects very rarely fail completely or get canceled. Often releases are delayed or released with "quality and usability issues." In the software world, this means that it will work, but not completely as planned – if it was a car, it might not have headlights and would occasionally shift into reverse when you turned on the windshield wipers. Customers may complain, but in most cases, the software will not be used as extensively as it was planned. It becomes "shelfware."

This is the end of our saga, which started with a poor estimate and the creation an of unrealistic project schedule and ended ignominiously. Now, let's try to understand how these mistakes were made. The root of the problem is in human psychology, which we will try analyze first.

The Way We Think When We Make Estimations

Let's go back to the original conversation between the programmer and the project manager. How long does it take to say, "I think abo-o-out … five days"? Try to do it. It shouldn't take longer than four seconds even if you pause before the sentence. Remember, we assume the programmer already understands the scope of the task. He uses these four seconds to make an estimate, here is his mental process:

1. *The programmer recalls the scope of the task. He does this very quickly as he has already thought about it. In his mind, he has already broken the activity into few small subtasks and has analyzed the duration of each subtask separately.*
2. *The programmer tries to recall similar projects, mostly with comparable scope or similar writing tools. He checks how relevant these projects are to the current task. That gives him an approximate answer, around four- five days. The programmer does this analysis almost instantaneously. In the programmer's mind, it is not an analysis; it is almost intuitive response based on his memory input.*
3. *The programmer spends the rest of these four seconds trying to come up with an acceptable estimate. Too low, he won't be able to complete his job on time. Too high, the project manager will be unhappy (he is not an idiot and has his own estimate).*

When the project manager asks "Are you sure?" the programmer doesn't repeat his analysis, he is just trying to determine whether his answer has satisfied the manager. Moreover, as you might have noticed, he did not have the time or information to understand any potential pitfalls, which he implicitly acknowledges with the phrase: "…if everything goes well…"

It is truly amazing how much thinking can be done in only four seconds. It is even more amazing that estimates done this way may be absolutely correct. Intuitive estimations work well when the person who performs the estimation has participated in similar activities many times and remembers his or her experiences very well. However, in research and development projects or, almost all new projects, this is often not possible.

Accurate estimation cannot be done without valid inputs. The information for estimations can only come from two sources:

1. Historical data or data about previous similar projects. This data can be captured in the project manager's brain or obtained as a result of data analysis.
2. Measurements of the current project's performance. Analysis of what has already occurred in the project makes is possible to forecast what will happen in the future.

The human brain processes this information extremely fast using certain simplification techniques. In many cases, they work very well, in others they can lead to systemic mistakes or biases.

Biases in Estimation

Let's go back to our example. When the programmer is attempting to find an acceptable answer for the project manager (see the last item in the programmer's thinking process), he gives a biased answer, in which he is motivated to come up with the comfortable timeline for himself. This is called **motivational bias**.

In addition, estimates can be influenced by external factors, such as release dates that can be another source of motivational biases. We know what we want to achieve, and we make our estimates to make this goal achievable. In our example, the project manager has a project with five activities which must be completed in five weeks for a release date. When he first starts to build the project schedule, he sets a duration of one week for each activity, in this way he will be able to deliver the project on time. He knows that his estimates are probably not accurate, but he will adjust them after his discussion with his team. But due to the release date, he will be motivated to accept any estimates that will not upset his plan.

So we see how the actual estimates are being skewed because of bias. Motivational biases are often easy to identify, but difficult to correct. They are like a disease: you know that you have flu, but cannot do very much about it rather than wait until it runs its course. In our example, the motivational biases of both of our actors can compound themselves. The programmer's biased estimate meshes nicely with the project managers motivational bias of completing the project on time. As the programmers estimate won't disturb the planned release date, it is placed in the schedule without and real scrutiny

Another type of mental error is called a **cognitive bias**. In 2002, Daniel Kahneman was awarded the Nobel Prize in economics for "having integrated insights from psychological research into economic science, especially concerning human judgment and decision-making under uncertainty." Interestingly, the Nobel Prize in economics was awarded to a psychologist rather than an economist. Kahneman's research significantly changed our understanding of human behavior, not only in economics, but also in other areas including project management.

According to the theory he developed together with Amos Tversky, people use heuristics or "rules of thumb" as a basis for their judgments. These heuristics are essentially certain simplification strategies or mental "shortcuts". In many cases, these heuristics work reasonably well. However, in some cases they provide a predictable faulty judgment or cognitive bias.

One of such "rule of thumb" is the **anchoring heuristic**. Again, let's return to our original example of estimation. The programmer tries to recall his previous projects to help estimate the duration of the current activity. He instantly comes up with a number, for example five days. Then, he compares this estimate to the other projects to determine if five days are sufficient. He may start to think that the task duration can be between four and six days or between three and seven days, but his analysis will always remain close to his original estimate of five days. The problem is that five days might be completely wrong, but it will "anchor" all future analysis.

Another type of mental shortcut is the **availability heuristic**. To explain how the availability heuristic can affect our estimates, let's do a very simple psychological exercise.

1. Take three seconds to try to recall all of the projects or large activities you were involved in the last year.
2. Now repeat step 1, but take 15 seconds
3. Repeat steps 1 and 2 taking 2 –3 minutes for each step and write down the results.

You will find that you will have a difficult time remembering more than a few previous activities unless you spend some time to think about it and you will probably have a clearer memory of your most recent projects. It is also easier to recall your most successful or largest projects. Interestingly, in our example, the programmer does not perform actual time calculations; instead, based on his previous experience, he attempts to understand the likelihood that the task duration will be five days. During these few seconds, the programmer spent would have been able to recall a very limited number of his previous activities. Therefore, he is making his assumption based on very limited dataset. He calculates the likelihood of the task taking five days based on the projects he remembers and if he remembers only his successful activities, he will underestimate the duration.

To illustrate, below is the set of his previous activities relevant to his current task of developing a computer program to display a bar chart:

Date	Activity	Clearly Remembers	Duration
Q1, 2001	Pie chart	No	10 days
Q2, 2003	Interactive bar chart	No	12 days
Q1, 2004	Multiple line chart	No	7 days
Q2, 2004	Small bar chart	Yes	3 days
Q4, 2005	Bar chart	Yes	5 days

Since at the time of estimation the programmer only clearly remembers two out of the five activities, he deems it very probable that the activity will be completed in five days. In reality, it could take much longer.

Another interesting phenomenon in estimation is the **Rule of PI** - regardless of how we do our estimations, we always underestimate, even if we are aware of the tendency to underestimate. Sound strange? Not if you think about it. Why are we repeatedly late and running out of time and money? We try to fit in too many activities into the project and hope against all hope that we will be successful. This wishful thinking is often the cause of the problem.

You can ask: Why PI? First, it emphasizes the fact that mistakes in estimations can be very significant. Second, this rule was invented by programmers who like to remind everybody that they know math.

Another interesting psychological phenomenon was mentioned by E. Goldratt in his Theory of Constraint. He calls it the **Student syndrome**. It refers to the normal student style of preparations for an exam, which leads to the wasting any contingency buffers built into individual task duration estimates. A good example is how we saw that our programmer really concentrated on the job only when the deadline was looming.

Example of Estimation

So we have seen how estimation is done in small projects with only a few players. But what about a large project? The problem is people’s mental machinery is the same regardless of the magnitude of the project.

Here is a story about a bridge over the Moscow River, close to Moscow City Center. The bridge by itself is quite remarkable, not only because of its architecture and location, but also because of magnitude of bad decisions that were made both during the original construction and later during the rebuilding. It is a dual subway/highway bridge with a subway station located on the lower deck of the bridge. The bridge connects Moscow’s South West with Moscow City Center.

The bridge was completed in 1959 in 15 months. During its construction, somebody decided to accelerate the hardening of the concrete in the main structural arcs by adding salt. It is hard to discover why and by whom such seemingly small but catastrophic decision was made, but soon after the bridge was completed, engineers noticed a fast corrosion of the reinforcement steel, which was due to the salt. A decision was made to try and cosmetically repair the bridge. The cosmetic measures were unsuccessful and in 1984 the subway station closed. However, highway on the upper level continued to operate.

The managers then decided to repair the bridge by replacing the major support structures while allowing the highway traffic to continue. How long would it take? It was difficult to estimate as the task was very complex, and some tasks that would be required had never been done before. The original estimate was nine years.

Apparently, the original estimate was biased. The availability heuristic definitely played a role as there was no historical data on which to base the estimates, the activities related to the disassembling of pre-stressed reinforced concrete structures had never been attempted before. It is quite possible that motivational biases were involved as well.

The reconstruction project started relatively fast, but later slowed significantly down due to technical issues. In the early 1990's construction almost halted due to budgetary problems related to the collapse of Soviet Union. Only in 1999 did the situation improve when another highway bridge nearby was completed. Traffic was rerouted to the new bridge and the old bridge was closed and almost completely demolished. The rebuilt bridge, which replicated the original design, was completed in 2001 and has been reopened to traffic. The subway station reopened in 2002. The rebuilding project took 18 years, twice the length of the original estimate of nine years.

This story teaches us few important lessons. First, many uncertainties, both technical and budgetary, were not properly captured in the analysis, which skewed the estimate. In addition, the mistakes in estimations caused an incorrect strategic decision. In hindsight, it would have been easier to build new highway bridge in another location rather than attempting to repair the existing bridge without disconnecting the traffic. However, because the original nine-year estimation looked reasonable, the managers did not select the new bridge scenario.

Simple remedies

Modern project management techniques offer a number of methods and tools that you can use to improve your estimations. But first, you need to try to avoid common mental traps that can occur while doing estimations.

One of the most important questions to ask is how can we integrate information about risks and uncertainties into our estimations? It is very difficult to analyze the effects of risks and uncertainties without specialized methods and tools, so at this point we'll concentrate on estimation based on "best case scenarios" as well as collected information about risks.

<u>Never Do A Wild Guess</u>

Estimations are possible based on partial information; however, we often try to make estimations without any or very little information. We will call this type of estimation a "wild guess". (For those of you who dislike the word "wild", we can also refer to it as an "intelligent guess"). For example, how much would is cost to develop one medication to treat all forms of cancer? There is no reliable information to support any answers to this question. However, these types of questions are often asked. We try to answer them, either because we don't want to look incompetent, or because management is really pushing us. The manager's position is quite understandable. He does not want to end up with question marks on the project schedule.

Unfortunately, as soon as we deliver the estimate, everybody instantly forgets that it was a "wild guess" and according to the anchoring heuristic, this estimate becomes the anchor for all future discussions. Could you imagine this newspaper headline: "Project manager of PharmaCo Inc. estimates that universal cancer drug will cost five billon dollars". Inevitably, people will use this number as starting point in any future analysis and discussions.

But what should we do if we are asked to make estimation without any information? The only solution is to get information from somewhere. If previous relevant data is not available, the only solution is to try some small task before starting a major activity and see what happens: how long will it take and what level of resources will it require? For example, you can make a prototype or evaluation tool. Unfortunately, sometimes management wants to forego this strategy and asks for an estimate immediately. This is where the big problems begin.

Collect Relevant Historical Data

Most project managers know how important it is to collect and analyze historical data related to previous projects, but very few actually do it. If we had full set of relevant activities in front of us, our estimates will be more accurate. In some industries, this data is available through various software applications, forms and methodologies. In other industries, using these tools does not guarantee accuracy. If you are lucky, you work for those organizations that routinely collect and analyze historical data as part of a portfolio management process.

But what if you don't have any tools and want to make accurate estimation? The simple solution is to keep your old project schedules handy, so you can easily access and review them when you are trying to make estimations.

Here is the simple way to analyze your information:

1. Look at previous project schedules or try to recall similar activities.
2. Write down activities and their relevance to the current one:

	Activity	Duration	Relevance
1	Development of UI for customer support software	20 days	relevant
2	Web site development	32 days	Not very relevant
3	Charts business analysis software	10 days	Almost the same
4	UI improvements for selected client	5 days	Relevant

3. Use this table to assess the duration of the current activity.

Proper collection and analysis of relevant historical data will help mitigate the negative effect of both motivational and cognitive biases, and will specifically help you address the availability heuristic.

Reality checks

Reality checks are a simple way to improve the accuracy of your estimations. The objective is to compare your estimations with known project results. Here is an example of how NASA analyzed the cost of the LISA project:

.

Einstein's theory of relativity has predicted the gravitational wave. NASA and European Space Agency are planning a mission called Laser Interferometer Space Antenna (LISA). LISA will consist of three sciencecraft, which will carry laser devices to measure passing gravitational waves. The launch is planned for 2011. The mission cost is very substantial and at the same time uncertain. NASA has used a number of techniques to estimate the cost. For example, NASA has plotted LISA against other NASA missions with similar size and cost.

It as a type of reality check. The analysis has shown that cost calculation for LISA project has been done consistenly with the other missions.

The main purpose of reality checks is to ensure that similar projects or activities will not require significantly different resources.

Independent Assessment

Independent estimations by different members of the same team or by different teams are a double-edged sword. On one hand, they offers additional look on the estimated parameters. On the other hand, psychologists know that assessments made by separate individual teams may be completely different and difficult to reconcile. Sometimes the independent assessment is as biased as the original estimate.

For some projects, especially with huge budgets, independent assessment is a requirement.

For example, for LISA Project, cost of the mission was independently assessed by JPL. JPL's cost estimate was $843m, while the original LISA Project cost estimate was $872m. Most importantly, after the independent assessment, an analysis was performed to discover what was the cause of the difference. This analysis found that the internal LISA Project estimates put more emphasis on technology development, which caused an increase in cost.

Sometimes an independent assessment may not be practical, as it may be difficult to find independent experts familiar with the particular project. Nevertheless, any discussions regarding the results of estimations between team members or independent teams have proven to be very effective as it helps to incorporate more information into the estimations.