**Guidelines of Software Project Estimation**

# I. Purpose

This document is general guidelines for software estimation and use results in efficiency managing software projects and continuously improving software processes.

# II. Software Estimation Process

## 1. Process Overview

The software estimation process begins with the definition of the project's functional requirements. Without defining the requirements, it is impossible to determine the cost and schedule for the software project. The old organizational process assets (OPA) should also be used to help calculate, validate and/or validate software estimates. Estimator should study the similar projects, size, effort, schedule, productivity, and quality of the historical project and adjust the estimation.

The next step in the process is to develop the size estimate. There are several methods are accepted for size estimation: Function Point (FP), Use-case Point (UCP), Work Breakdown Structure (WBS) and Line of Code (LOC). It is advisable to use more than one method to calculate size, so that results can be compared. Once the size has been determined, the next step is to calculate the effort, cost, and schedule of the project using the size estimate that was calculated previously. Effort is usually calculated in person months, which then can be translated into cost by using the applicable labor rates. Software effort, cost and schedule are all interrelated and a change to one will affect the other two.

Estimates of resources (both human resource and infrastructure resource) shall also be made. Items such as memory, processor, hard-disk and network requirement should be calculated and tracked. It is important to leave some reserve capacity in infrastructure resources for future software upgrades.

After the size, cost, schedule, and resources have been calculated, a risk assessment should be conducted. Risks associated with the project likely will cause the estimate to change. These risks should be documented, tracked, and updated over the life of the project. After the estimation risks have been identified and changes, if required, made to the previous estimate, the estimate then must be validated. The purpose of the validation is to ensure that assumptions made for the estimate are accurate, that the estimate itself is reasonable and accurate, the methods used to develop the estimate were appropriate, all risks have been identified, and to confirm and record the official estimates for the project. Verification is usually performed by management personnel or a quality assurance organization.

Software estimates should be continually tracked and updated throughout the life cycle of a project. Each time an estimate is updated, the assumptions and inputs shall also be updated to reflect the most current information.

A Software Estimation Record must be maintained that contains information such as, estimation methods used, date of estimate, size, cost, schedule, resources, and risks for each estimate that is developed. This is to upgrade the OPA, used for estimating in the future projects.

This process should be utilized throughout the life cycle of a project. It also can be implemented at any point in the project's life cycle. So, whether the project is in the development or maintenance phase, this process will provide added value.

## 2. Process for Software Estimation Activities

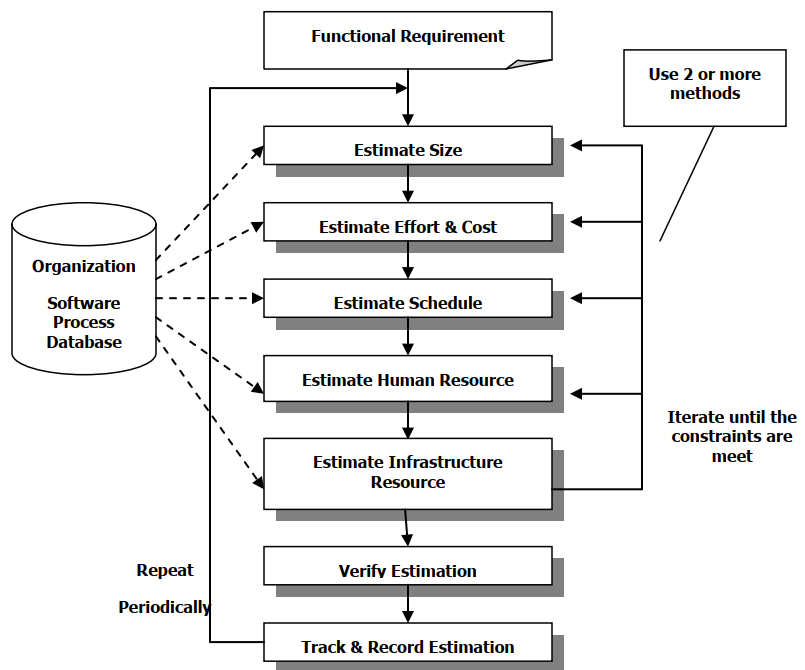
All steps in the estimation process are carried out by Project Manager.

Software estimation is a continual process that should be used throughout the life cycle of a project. The software estimation process consists of the following activities:

* Estimate size
* Estimate effort & cost
* Estimate schedule
* Estimate resources (HR, infrastructure)
* Verify estimation
* Track and record estimates

The process activities for developing the size, effort, and cost are shown before the schedule estimate in below figure because this is the sequence often used by the cost models.

However, a development schedule is often mandated before the scope of the effort is clearly understood. The early establishment of a WBS helps to divide the effort into distinct work segments that can be scheduled and prioritized.



### Step 1: Estimate Size

The size of the software is the predominant factor in determining how much effort is needed to build it. But the ultimate size is not known when the project is being conceived, and the software does not exist. Hence, if size is to be used for the effort estimation model, it must be estimated for the initial estimation.

Software functional requirements must be defined before size estimation can occur. Project size can be measured by function points or use-case points or functional WBS. Both of them are based on the functionality of a system and can be applied during the Definition stage or earlier. It is preferred that the estimation is carried on by both methods, so that the result can be verified easier. If the standard estimations are not suitable with the projects, the PM could estimate size by developing his method or using other methodologies like LOC calculation, etc.

Detail guideline of Size estimation using various methods is available in WBS Estimation Guideline

Regarding to LOC estimation, it is done by experienced people and historical data. There is no documented methodology/guideline available for this estimation.

### Step 2: Estimate Effort & Cost

A common approach is top-down approach that uses the size estimate together with the historical data of productivity from the OPA to derive the total estimated project effort. This equation can be determined through regression analysis of past data on effort and size. Then, once the overall effort for the project is known, the effort for various stages and activities can be determined as a percentage of the total effort.

In another approach named bottom-up, you obtain the estimates first for parts of the project and then for the overall estimate. That is, the overall estimate of the project is derived from the estimates of its parts. One bottom-up method calls for using some type of activity-based or WBS estimation. In this strategy, the major activities are first enumerated, and then the effort for each activity is estimated. From these estimates, the effort for the overall project is obtained. A good way of developing your project’s WBS is using the major milestones/deliverables that you identified in the Work Order. Once, the milestone has been identified, brainstorming with your project team to detail the tasks that you must accomplish in order to achieve the milestones.

The bottom-up approach lends itself to direct estimation of effort; once the project is partitioned into smaller tasks, it is possible to directly estimate the effort required for them. Although size does play a role in determining the effort for many activities in a project, a key advantage of this approach is that it does not require explicit size estimates for the software. Instead, it requires a list of project tasks, which might be easier to prepare in some situations. A risk of bottom-up methods is that you may omit some important activities in the list of tasks.

Both the top-down and the bottom-up approaches require information about the project: size (for top-down approaches) and a list of tasks (for bottom-up approaches). In many ways, these approaches are complementary. Both types of estimates are more accurate if more information about the project is available or as the project proceeds. For example, estimating the size is much more difficult when very high level requirements are given but becomes considerably easier when design is finished, and even easier and more accurate when code is developed. Thus, the accuracy of estimates depends on the point at which effort is estimated, with accuracy increasing as more information about the project becomes available.

The results of the two estimates should be compared, and reasons for any large variances should be resolved. How large a variance is acceptable depends upon where a project is in the life cycle. Early in the project, estimates within 20% are reasonable. As a project matures, the estimates should converge. However, if spread is bigger than 20%, a detail investigation should be carried on to reduce the gap. This comparison method should also be used when calculating size and schedule estimates. All assumptions and inputs should be documented in the estimation record.

If accepted by client the result of effort estimation better keep in as a range. This range will be reduced toward the end of project life-cycle. If the firm estimation is required, the final estimation result should be taken as an average result from both estimations.

Cost estimation will be derived from the effort estimation result by multiplying the last with average rate per man-month or man-day.

### Step 3: Estimate Schedule

Once the effort is known or fixed, various schedules (or project duration) are possible, depending on the number of resources (people) put on the project. However, effort and months are not fully interchangeable in a software project. For instance, no one would execute the project in 28 months with 2 people. In other words, once the effort is fixed, you can gain some flexibility in setting the schedule by appropriately staffing the project, but this flexibility is not unlimited.

The procedure for project schedule estimation should follow the bellowed steps (details would be shown in the lesson of Creating Project Schedule)



You have learned how to flesh out the activities involved in a project, determine the sequence of the activities, and establish a schedule. You are now ready for the next step – determining the resource needed to accomplish project activities. Note that once you have determined the resource and their availability, you may need to review your schedule (Gantt chart).

### Step 4: Estimate Human Resource

You will need to determine the skills required to accomplish the activities. After this phase, you need to match people to those skills. A good way is to create skill sheet that match the skills to the activities – this information can sometimes be obtained at the same time. Often the manager of the people you need will be able to commit the availability of these people to your project. However, it is recommended that you always cross-check that information with your team member, and pay attention to the following.

* Check up the availability of your team members by taking into account their vacation, sick days, and the other projects that they are already working on
* Ask the functional manager of your people to assess the skill set of your people and the effort days that will be required by them to accomplish the activities assigned to them (always cross-check with the effort days with the person who will executing the activity)
* The functional manager should be able to tell you how much their people cost. The more detail you go into when estimating the costs, the more accurate your cost estimates will be. So be detailed when it comes to costs as this frequently become an issue later on the project.

### Step 5: Estimate Infrastructure Resource

Besides human resource you may also need to factor in the availability of non-people resource such as equipment, facilities. To do this, you shall create Infrastructure plan.

To estimate the infrastructure resource, several methods are used: historical experience, simulations, prototyping, and analysis. The method used for the calculation of infrastructure resources is project-specific. Infrastructure resources include items such as, memory usage, throughput capacity, timing, and hard drive usage. Different projects have different infrastructure resources, so projects may estimate and track different infrastructure resource items. It is important to plan for the availability of excess infrastructure resources in anticipation of future program updates.

Infrastructure resources may be in the host environment, in the integration and testing environment, in the target environment, or in any combination of these.

Examples of infrastructure resources include:

* Computer memory capacity.
* Computer processor use.
* Communications channel capacity.

The steps for estimating the project’s infrastructure resources as following:

1.Infrastructure resources for the project are identified by the Project managers.

2.The Project managers estimate for the infrastructure resources in consistence with the estimates of:

* The size of the software work products.
* The operational processing load.
* The communications traffic.

The activities for managing the project's infrastructure resources follow the bellowed steps:

1. Estimates for the project's infrastructure resources are derived based on historical experience, simulations, prototyping, or analysis, as appropriate.

* Sources and rationale for estimates are documented.
* Similarities and differences between the project and the sources for historical data in
* terms of application domain and design approach are assessed and recorded.
* The reasoning used to judge the credibility of the estimates is recorded.

2. The planned computer resources, the system requirements allocated to software, the software requirements, and/or the software design are adjusted to achieve the project's infrastructure resource requirements.

3. The available computer resources are allocated to the software components.

4. The available capacity for the infrastructure resources provides for a specified reserve capacity when the initial estimates are made.

Now that you have idea of your resource requirements and availability, go back to your Gantt chart, you might need to re-adjust it in order to take into account your team members or other non-person resource’s availability.

Note that in case the budgeted effort and schedule are already given, the process steps 1-5 can be iterated as many time as required to get the nearest possible schedule and effort with the budgeted ones.

### Step 6: Verify Estimation

The purpose of this step is to verify quality of the estimation and to get it approved

* Confirm the software architecture and functional WBS
* Verify the methods used for deriving the size, schedule and cost estimates
* Ensure that the assumptions and input data used to develop the estimates are correct
* Ensure that the estimate is reasonable and accurate given the input data
* Formally confirm and record the official estimates for the project.

### Step 7: Track and Record Estimates

Estimates are tracked over time. The frequency of re-estimation at milestone or event-driven depend on the releases of critical project deliverables (when the level of project uncertainty or project size is changing). For example, after completion of SRS, the project size should be re-estimated. Same thing must be done after design complete, code complete, or receiving change requests. At the end of project, in post mortem report, the project size should be re-estimated again. Based on actual effort spent, the productivity for project can be calculated.

Then, estimation data, both planned and actual, are included in OPA to maintain the historical database of estimates. This historical database can be used by the estimator to either calibrate cost models or for purposes of comparison when performing estimates for future projects

# III. Work Breakdown Structure Estimation

## 1. Overview

Effort and schedule estimation are essential activities in the project planning stages. This document defines how effort estimates can be produced for a project. The general approach to estimation is to estimate the effort first, and then use it, along with other factors like available resources, etc. to estimate the schedule.

There are two fundamental approaches to estimation, including top-down and bottom-up. In a top-down approach, the overall estimate for the project is first determined based on some models and then the estimates for different smaller tasks are determined. Such a model typically requires, as input, the size estimate (in Lines of Code -LOC or function points or use case points) of the overall system.

The bottom-up approach, on the other hand, first defines the various activities that need to be executed for the project. As these activities are typically at a sufficiently low level of granularity, effort estimate of each activity can be done from past experience. The estimate for the total project is then obtained from the estimates of these activities. By the nature of this type of activity based estimation, there is no need to estimate or track the sizes of programs or systems. But for bottom-up estimation, sufficient experience with similar kinds of projects or applications is required to carry out the task. Milestone analysis of projects already executed can provide the information required.

Data from OPA is suggestion for estimation. Estimator should study the similar projects, size, effort, schedule, productivity, and quality in the OPA to measure and adjust the estimation.

## 2. WBS Estimation Methods

In this method, the project work is first divided into a hierarchy of tasks (or units), which is called Work Break-down Structure. Each task will then be estimated individually. The estimated effort of the whole project is the aggregation of the effort required for each task. (So, these methods belong to bottom-up approach.) The estimation for the lowest level of decomposition is usually assigned to more than 1 person. If the results they got are different from each other, one of methods listed in Standard methods used in manual estimation section can be used to make them converged.

There are 2 approaches to WBS estimation. The first one is based on program-module decomposition, which is called Module-based WBS estimation and the other is based on project’s activities, which is called activity-based estimation.

The level of detail in the work breakdown should be driven by the level of accuracy needed for estimating resource and duration. However, work break down should include no more than six levels, and only very large projects require six levels. Projects that require more levels should be broken into subprojects. Normally, projects need only three levels. The work breakdown is considered controllable when the work packages at the bottom level are in range of 0.5 – 2% of total project effort.

The template is available in Template\_Estimate WBS.xls

### a. Module-based WBS estimation

The whole project is divided into modules or programs (according to the proposed architecture). Each program is classified as simple, medium, or complex, and the build effort for each program is estimated based on past experience of similar projects. The effort for other stages of the project is estimated using the effort distribution on similar projects. Some guidelines for classifying into simple, medium, complex (S/M/C) are given later in this document. Notes: the effort spent for support activities (like project management, configuration management...) are bundled inside the effort estimated for each program.

The estimation is performed when the requirements are clear, or have been approved. The procedure for estimation has the following steps:

* Step 1: Identify programs in the system and classify them as simple, medium, or complex (S/M/C).
* Step 2: If a project-specific baseline exists, get the average build effort for S/M/C
* from the baseline.
* Step 3: If no similar project exists (i.e. this project is in an area/technology in which there is no experience), then use the average build effort from existing projects with a contingency add-on effort to take care of the risk.
* Step 4: Use project specific factors to further refine the build effort for S/M/C. Obtain the total build effort using this and the counts for S/M/C.
* Step 5: Use the effort distribution data of similar projects, to estimate effort for other tasks and the total effort.
* Step 6: Refine the estimates based on project specific factors.

### b. Activity-based WBS estimation

The whole project is divided by tasks (or work package) based on certain criteria. In this decomposition, the software engineering tasks and support processes (like project management, configuration management, QA...) are separately estimated.

The decomposition of work packages for each level in WBS hierarchy is implemented according to one of the following methods:

* Based on system architecture (defined in architectural design).
* Based on process model (ex: requirement definition, analysis, design...)
* Based on project life-cycle/milestones (ex. definition, solution ...)
* Based on specific locations (for deployment projects).

The estimator can combine different levels in the work breakdown structure. For example, the system architecture based-method is used in the highest level of work packages but lower level ones are defined by the process model based method.