

GALORATH

10-Step Estimation Process Checklist

www.galorath.com

Step 1: Define Scope and Purpose

Estimation begins with clarity about what is being analyzed and why. Establish the boundaries, goals, and contacts before data collection begins.

ITEM	NOTES / DESCRIPTION
Identify scope	Specify whether the estimate is an EAC, ETC, or other type. Note if integration with other systems is included and list all related programs.
Define purpose	State the primary intent of the estimate such as cost analysis, schedule projection, or staffing plan.
Establish points of contact	List individuals responsible for technical data, contract information, and requirements support.

Step 2: Establish Technical Baseline, Ground Rules, and Assumptions

A defined baseline sets the foundation for accuracy. Capture every assumption and condition influencing the model.

ITEM	NOTES / DESCRIPTION
Describe technical baseline	Outline each CSCI, its relationship to other components, and any planned builds or deliveries.
Record ground rules	List known conditions or mandatory constraints that shape development.
Record assumptions	Document information not yet available but required to complete the estimate.
Align structure	Ensure the baseline aligns with the WBS used for reporting and analysis.

Step 3: Collect Data

Reliable estimates depend on complete and validated input data.

ITEM	NOTES / DESCRIPTION
Customize data worksheets	Pre-populate parameters where possible to reduce time spent with stakeholders.
Complete data sets	Fill out the collection forms for each CSCI or equivalent component.
Review for accuracy	Check parameters for reasonableness and internal consistency.
Verify uncertain values	Confirm questionable entries with developers or contractors.

Step 4: Determine Software Size

Size drives cost, effort, and duration. Define it precisely for new, reused, and modified code.

ITEM	NOTES / DESCRIPTION
Estimate new software size	Record least, likely, and most values with clear definitions of SLOC or Function Points.
Capture reused code	List all pre-existing software and note actual or estimated size.
Evaluate rework effort	Identify adjustments needed for modified or legacy components.
Identify COTS and prototypes	Mark all commercial or prototype code used.
Apply growth factors	Include expected size growth throughout development.

Step 5: Prepare the Baseline Estimate

Create a traceable foundation for all future comparisons.

ITEM	NOTES / DESCRIPTION
Enter collected data	Input verified parameters into the estimation model.
Document inputs	Include concise notes for each parameter, even if default knowledge bases are used.
Review inputs and outputs	Confirm that entered data produce results consistent with expectations.
Conduct joint review	Compare analyst and developer perspectives and document any differences.
Compare baselines	Record changes between this estimate and previous versions.

Step 6: Quantify Risks and Analyze Impact

Recognizing uncertainty improves decision quality.

ITEM	NOTES / DESCRIPTION
List development risks	Identify risks for each CSCI or software element.
Select risks for inclusion	Indicate which risks are modeled within the estimate and why.
Define communication method	Decide whether risks are presented collectively or individually to show potential impact.

Step 7: Validate and Review the Estimate

Ensure that the estimate is credible, explainable, and aligned with organizational capability.

ITEM	NOTES / DESCRIPTION
Check productivity	Confirm that estimated productivity aligns with benchmarks and team performance.
Verify staffing	Assess whether projected staffing can be met with available resources.
Review size and growth	Ensure size assumptions are realistic and include growth allowances.
Evaluate cost drivers	Confirm top drivers reflect actual program priorities.
Check schedule alignment	Verify that project timelines fit within master schedules.
Confirm risk inclusion	Make sure risks and confidence levels are documented.
Validate cost inputs	Recheck labor rates and included cost elements.



Step 8: Build the Project Plan

Translate the validated estimate into a practical execution plan.

ITEM	NOTES / DESCRIPTION
Generate plan	Build the project schedule, milestones, and staffing plan using baseline data.
Verify alignment	Ensure plan assumptions match the validated estimate.

Step 9: Document the Estimate and Lessons Learned

Comprehensive documentation supports traceability and future reuse.

ITEM	NOTES / DESCRIPTION
Record parameter rationale	Provide justification for every selected input.
Maintain version control	Label and archive each iteration of the estimate.
Attach supporting materials	Include ground rules, assumptions, and reference data.
Capture unique circumstances	Note conditions that influenced the estimate’s scope or focus.

Step 10: Track and Update During Development

Estimation remains valuable when maintained throughout the program lifecycle.

ITEM	NOTES / DESCRIPTION
Collect actual data	Use standard collection forms to record performance metrics.
Update baseline	Enter current data to maintain visibility into cost, schedule, and staffing trends.
Report results	Share updated metrics and insights with stakeholders.

Checklist Use

This 10-step checklist provides a repeatable framework for producing transparent and defensible estimates. Use it as a working reference to strengthen planning discipline and maintain alignment between projected and actual performance.

The Software Cost Estimation eBook explains the reasoning, models, and examples behind each step. [Download the eBook](#) to learn how to apply these methods across your organization’s estimation practices.