# Managing Schedule Risk Expectations During Program Execution

**Patrick K. Malone**
**Systems Planning and Analysis, Inc.**
**550 Continental Blvd. Ste 185**
**El Segundo, CA 90245**
**pmalone@spa.com**

*Abstract*— **Program managers use Integrated Master Schedules to complete projects. For large programs it provides a detailed time phased logical view of required work using the critical path method with a corresponding logical activity network. The Government Accountability Office's 19th annual assessment of the Department of Defense's weapon programs indicates there have been significant changes to the department's acquisition process to "deliver solutions and capability to the end user in a timely manner". However, they found programs have acquisition approaches that still result in schedule and cost challenges similar to those previously reported. One factor embedded within the process is success driven schedules that minimize major potential setbacks like design flaws, test failures, and optimistic task durations. When incorporating risks, they are often treated as a separate activity turned into a cost impact and not directly attached to schedule impacts. Moreover, organizations generally identify risks using rule-based risk taxonomies; selecting risk exposures and impacts from tables grading them low to high. Assessments are carried forward but not implemented in the schedule. This Paper investigates schedule risk methods and modeling techniques to provide an approach that more realistically forecasts schedule impacts. The resulting process shows how to implement multiple risks across programs.**

## TABLE OF CONTENTS

## 1. INTRODUCTION

Program managers use schedules and lists to manage their projects to completion. For large programs there is generally an Integrated Master Schedule (IMS). It provides a detailed time phased logical view of all required work scope across integrated product teams (IPT's) using the critical path method (CPM)[1] and a corresponding activity network. A program is often driven by the due date to meet a critical product introduction, initial operational capability, or launch window.

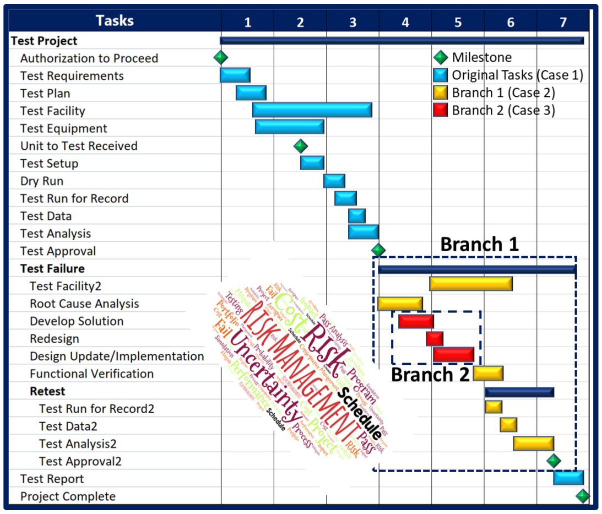


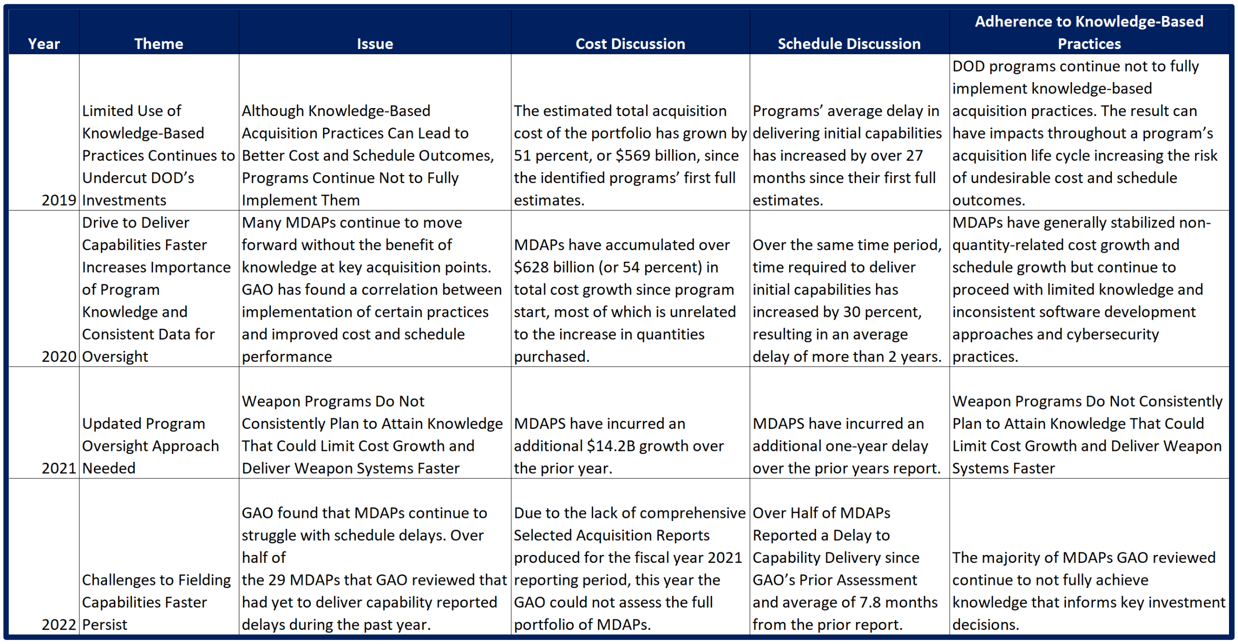
**Figure 1. Integrating risk into schedules support realistic forecasting**

The Government Accountability Office's 19th annual assessment of the Department of Defense's (DOD) weapon programs (Published June 2021) states there have been significant changes to the department's acquisition process. A key element of the departments changes was to "deliver solutions and capability to the end user in a timely manner". However, they found many programs have acquisition approaches that still result in cost and schedule challenges similar to those previously reported. At last count between 2019 and 2021, cost increased another 54% with schedule

---

[1] The critical path method is a process to determine longest stretch of dependent activities and time required to complete them from start to finish.

**Table 1. GAO indicates programs still begin with limited knowledge leading to cost and schedule growth.**



| Year | Theme | Issue | Cost Discussion | Schedule Discussion | Adherence to Knowledge-Based Practices |
|---|---|---|---|---|---|
| 2019 | Limited Use of Knowledge-Based Practices Continues to Undercut DOD's Investments | Although Knowledge-Based Acquisition Practices Can Lead to Better Cost and Schedule Outcomes, Programs Continue Not to Fully Implement Them | The estimated total acquisition cost of the portfolio has grown by 51 percent, or $569 billion, since the identified programs' first full estimates. | Programs' average delay in delivering initial capabilities has increased by over 27 months since their first full estimates. | DOD programs continue not to fully implement knowledge-based acquisition practices. The result can have impacts throughout a program's acquisition life cycle increasing the risk of undesirable cost and schedule outcomes. |
| 2020 | Drive to Deliver Capabilities Faster Increases Importance of Program Knowledge and Consistent Data for Oversight | Many MDAPs continue to move forward without the benefit of knowledge at key acquisition points. GAO has found a correlation between implementation of certain practices and improved cost and schedule performance | MDAPs have accumulated over $628 billion (or 54 percent) in total cost growth since program start, most of which is unrelated to the increase in quantities purchased. | Over the same time period, time required to deliver initial capabilities has increased by 30 percent, resulting in an average delay of more than 2 years. | MDAPs have generally stabilized non-quantity-related cost growth and schedule growth but continue to proceed with limited knowledge and inconsistent software development approaches and cybersecurity practices. |
| 2021 | Updated Program Oversight Approach Needed | Weapon Programs Do Not Consistently Plan to Attain Knowledge That Could Limit Cost Growth and Deliver Weapon Systems Faster | MDAPS have incurred an additional $14.2B growth over the prior year. | MDAPS have incurred an additional one-year delay over the prior years report. | Weapon Programs Do Not Consistently Plan to Attain Knowledge That Could Limit Cost Growth and Deliver Weapon Systems Faster |
| 2022 | Challenges to Fielding Capabilities Faster Persist | GAO found that MDAPs continue to struggle with schedule delays. Over half of the 29 MDAPs that GAO reviewed that had yet to deliver capability reported delays during the past year. | Due to the lack of comprehensive Selected Acquisition Reports produced for the fiscal year 2021 reporting period, this year the GAO could not assess the full portfolio of MDAPs. | Over Half of MDAPs Reported a Delay to Capability Delivery since GAO's Prior Assessment and average of 7.8 months from the prior report. | The majority of MDAPs GAO reviewed continue to not fully achieve knowledge that informs key investment decisions. |

delays adding another 40% to initial operational capability (IOC) launch from over 27 months to almost 38 months. [1]

Subsequent analysis in the 2019 – 2022 GAO reports indicate these issues impact cost and schedule growth to field needed capability. This is rooted in starting Engineering, Manufacturing and Development (EMD) prior to attaining the required program knowledge outlined in the GAO knowledge point framework (Appendix A). They found the additional risk of unknown information not fully vetted in technology development did not fully achieve knowledge that informs key investment decisions. Compounding some of the growth and delays were after-effects of the COVID-19 pandemic. In their 2022 analysis, they found the results were mixed across programs but recommended surveying the industrial base health to attain additional insight. Table 1 is a summary of GAO reports from 2019 – 2022 that highlight the knowledge themes and impacts to cost and schedule.

Another factor embedded within the acquisition process is use of success driven schedules that minimize any major setbacks such as a systemic design flaw, test failure, optimistic task durations or other items that could impact an "on-time" product delivery. When program risks are incorporated in the project, they are often treated as a separate activity, not directly attached to the schedule. Furthermore, organizations typically use a rule-based risk taxonomy. That is, selecting a risk exposure from a set of tables that grade risks from low to high and the impact or consequence if it occurs, be it cost, schedule or technical performance. These subjective assessments are then carried forward as part of the risk process but not necessarily implemented in the schedule. While cost may be integrated into the program estimate at completion (EAC), schedule impacts are generally translated to cost without regard for the actual (added) tasks and resulting time impacts.

This paper investigates (schedule) risk integrated directly into the IMS to provide a more accurate representation of how the program might be executed (Figure 1). First, we show a simple example and resulting exposure. Then, show, through simulation, an evaluation of sensitivities and comparisons to this baseline example. This effort then presents simulations with probabilistic solutions that are higher fidelity than the rule-based method allowing decisionmakers early insight to implement corrective actions over the traditional rule-based method. This approach supports integrating multiple risk simulations in an IMS or schedule portfolio adding to the development of appropriate cost and schedule reserves which will follow.

Future work will develop optimization methods to minimize the effort needed to identify required simulations and still allocate cost and schedule margins.

## 2. TRADITIONAL RISK MANAGEMENT

### *Risk and Uncertainty*

There is an important distinction between the terms risk and uncertainty. *Risk* is the chance of loss or injury. *Uncertainty* is the indefiniteness about the outcome of a situation. In a situation that includes favorable and unfavorable events, risk is the probability an unfavorable event occurs [2]. Uncertainty is assessed in (schedule) models for the purpose of estimating the likelihood (probability) that a specific event or outcome might occur by its scheduled due date when task durations are completed earlier or later than the most likely duration. In most cases this is interpreted as a schedule risk assessment (SRA). However, it is really an assessment of execution uncertainty within a success driven schedule; not a schedule with defined risks due to execution anomalies. In many cases, a baseline schedule will contain one or more schedule margin tasks to account for these risk uncertainties. To show the difference, the baseline example modeling uncertainty is developed, then adding to the model, a possible risk event is added to compare the results. Using both (uncertainty and risk) shows how they differ. We emphasize how the probability of an unfavorable event occurs and the resulting impact, then what might be done to mitigate or minimize the impact if the event does occur.

Schedule activity durations should account for both risk and uncertainty. Risk and uncertainty in scheduling refer to the fact that because activity durations are forecasts (estimated based on history, expert judgement or engineering belief), there is always a chance that actual activity durations—and therefore scheduled start dates and finish dates—will differ from the plan. [3] This paper addresses a project schedule with an expected uncertainty. Then we build on it to show how incorporating risk with uncertainty will provide significantly different results.

*Department of Defense*

The United States Department of Defense (DoD) has a mature risk management process that addresses risks, issues and opportunities. Risks are future uncertainties relating to, and achieving program technical performance goals within defined cost and schedule constraints. Issues are current problems that should be addressed with action plans and resources to support resolution. Opportunities are events that may or may not occur that have the potential for improving the program outcome in terms of cost, schedule, and performance. [4]

The DoD risk methods are integral with the system engineering and program management disciplines and use a top-down approach to identification and mitigation. There is a focus on cost, schedule and performance goals to meet mission objectives through a well-structured and repeatable method. When implementing a risk for a specific program or portfolio, the DoD is encouraged to align itself with the prime contractors' methods to support a consistent assessment of all risks, issues and opportunities (both Government only and contractor).

*NASA*

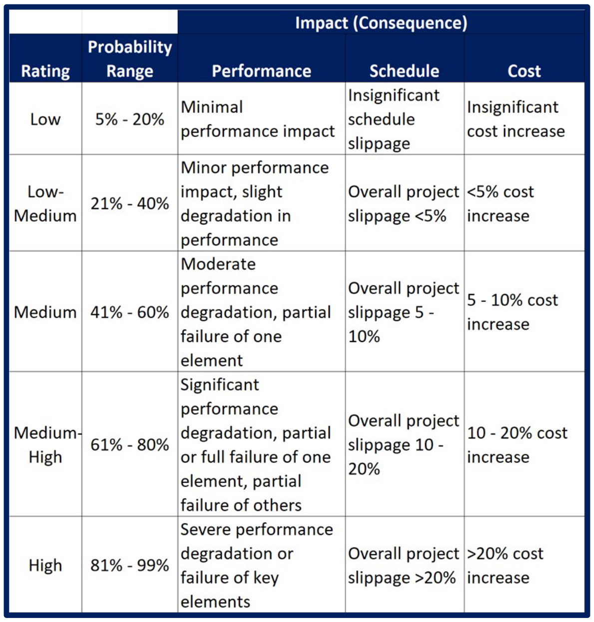
NASA since its inception, has implemented a qualitative risk management method. Initially, it was based on a set of rules, or rule-based risk management. This method was found to be inflexible and sometimes ineffective. [5] Next, the method now known as Continuous Risk Management (CRM) was implemented and more recently a rigorous quantitative risk management method called Risk-Informed Decision Making (RIDM) and an enhanced version of CRM is used. [6] NASA now integrates both methods in its handling of project, program and portfolio risks. The purpose of RIDM has matured and is based on lessons learned. The method addresses such things as mitigating "mismatches" between stakeholder expectations and the "true risks" (i.e., resources, processes, material, etc.). It is designed to achieve realistic expectations and minimize miscommunications when considering respective risks associated with competing alternatives. The approach begins with CRM and expands using probabilistic methods as part of RIDM. The results provide stakeholders and decisionmakers the ability to objectively evaluate proposed alternatives and fundamental risks regardless of the probability of occurrence.

*Industry*

Industry in general has built into their engineering and/or program management processes, a risk management component. When planning and executing projects or programs, there is generally a company specific rule-based risk taxonomy from which to draw guidance for implementing a qualitative risk assessment for technical performance, cost and schedule impacts. Sometimes a quality component is included.

In a Project Management Network article, Roger Graves states, "Risk assessment provides an estimate of the severity of a risk. [7] Without this assessment, a project manager can waste time on risks that may be of little importance to the project, or, worse, fail to give sufficient attention to significant risks." Further, he states a "detailed quantitative analysis of risks is always preferred, in many cases this is neither practical nor possible. Qualitative assessment of risks, however, can always be performed, and will usually take far less time and resources than quantitative analysis."

**Table 2. Common Rule-based" risk lookup table to assign risks**



| | | Impact (Consequence) | | |
|---|---|---|---|---|
| Rating | Probability Range | Performance | Schedule | Cost |
| Low | 5% - 20% | Minimal performance impact | Insignificant schedule slippage | Insignificant cost increase |
| Low-Medium | 21% - 40% | Minor performance impact, slight degradation in performance | Overall project slippage <5% | <5% cost increase |
| Medium | 41% - 60% | Moderate performance degradation, partial failure of one element | Overall project slippage 5 - 10% | 5 - 10% cost increase |
| Medium-High | 61% - 80% | Significant performance degradation, partial or full failure of one element, partial failure of others | Overall project slippage 10 - 20% | 10 - 20% cost increase |
| High | 81% - 99% | Severe performance degradation or failure of key elements | Overall project slippage >20% | >20% cost increase |

*Traditional Rule-Based Method*

Common qualitative risk management methods use *rule-based* "lookup" tables to assess risk. [8] Risks (and

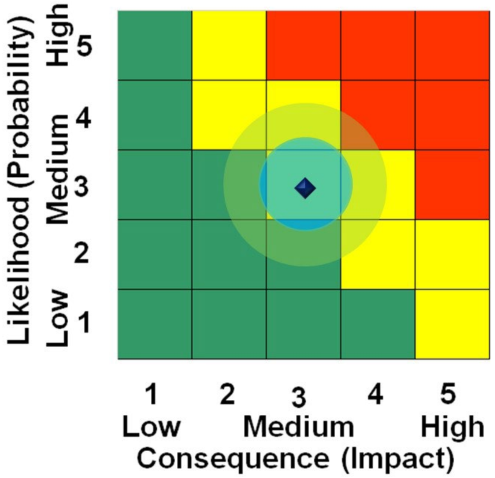


**Figure 2. Initial Rule based risk assessment**

opportunities) are developed and charted into a risk cube, generally a 5 X 5 matrix with the probability and impact based on the table data. This top-down method is adequate during initial project planning to "scope" the overall risk. Table 2 is an example from Graves, showing a typical risk taxonomy used for this purpose.

When using this traditional method, a risk is identified and then, using the Delphi approach or expert opinion, the risk is evaluated. Subsequently, a "burndown" plan, a set of activities designed to mitigate or avoid the risk are implemented. For this paper, a risk is identified to support the examples. A group or subject matter expert (SME) indicated there is a possibility of a test anomaly or failure. The risk is medium with a probability of 50% (a coin toss). Under the rule-based taxonomy of Table 1, schedule could slip 10% and cost growth could be up to 10% if the risk occurs. We will use this example to contrast simulation variants in the forthcoming examples. This is shown in the 5 X 5 of Figure 2.

## 3. SCHEDULE AND RISK INTEGRATION

When addressing schedule risk, there are four common methods to address risk and uncertainty. These are generally based on historical data. First an evaluation of the critical path is done using a deterministic network model that has a single duration estimate for each task. Then, integrating risk and uncertainty incorporating lessons learned and experience from prior programs to define impacts. The second method is a weighted average of durations using a triangular distribution method. The task durations are estimated using the sum of the optimistic, most likely and pessimistic durations divided by 3. (Equation 1) Third is referred to as the beta (PERT)[2] distribution. This method uses a weighted average with the most likely being four times that of the optimistic and pessimistic durations divided by 6 (Equation 2). [9]

$$\textbf{Duration} = \textbf{(OD + MLD + PD)/3} \qquad \textbf{(1)}$$

Where:
OD = Optimistic Duration
MLD = Most Likely Duration
PD = Pessimistic Duration
And the Beta (PERT) method

$$\textbf{Duration} = \textbf{(OD + 4*MLD + PD)/6} \qquad \textbf{(2)}$$

The fourth method is a Monte Carlo method where many simulations are run randomly selecting different durations for each task (within a specified range and occurrence) for each task. The result of this method is a probability density function (PDF) (and cumulative output) or "S"- curve that presents the confidence level of meeting a specified date.

Referring back to the traditional rule-based risk method in the prior section, tasks that have been identified as risk elements from the medium risk, schedule for those tasks would incorporate a 10% penalty in duration that is added to the deterministic or pessimistic durations. In addition, integrating risk into the project schedule provides better cost insight to quantifying risks. [10] All of these methods are valid approaches to account for uncertainty and even some risks. However, they do not necessarily incorporate the existence of risks or alternate courses of action that are likely needed to more accurately model risk scenarios. If activities do not go

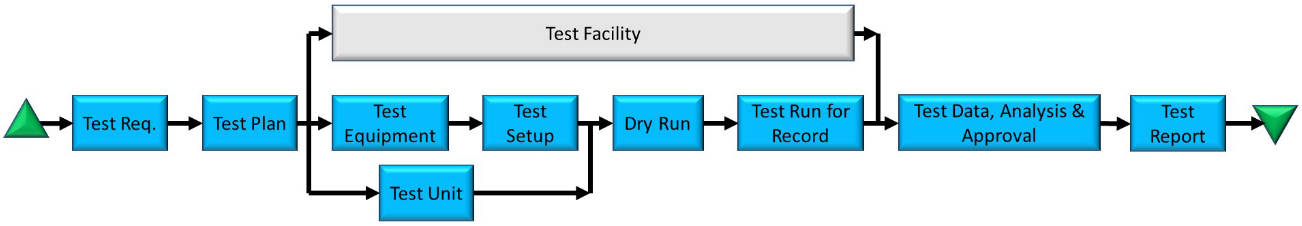


**Figure 3. Baseline Case 1 (Success driven)**

---

[2] The program (or project) evaluation and review technique (PERT) is a probabilistic or stochastic network model that uses three duration estimates. When using this method, the activity duration is still an estimate of the time required to perform the work within a task activity. It should consider the nature of the task activity, productivity impacts and the nature of the identified risk(s).

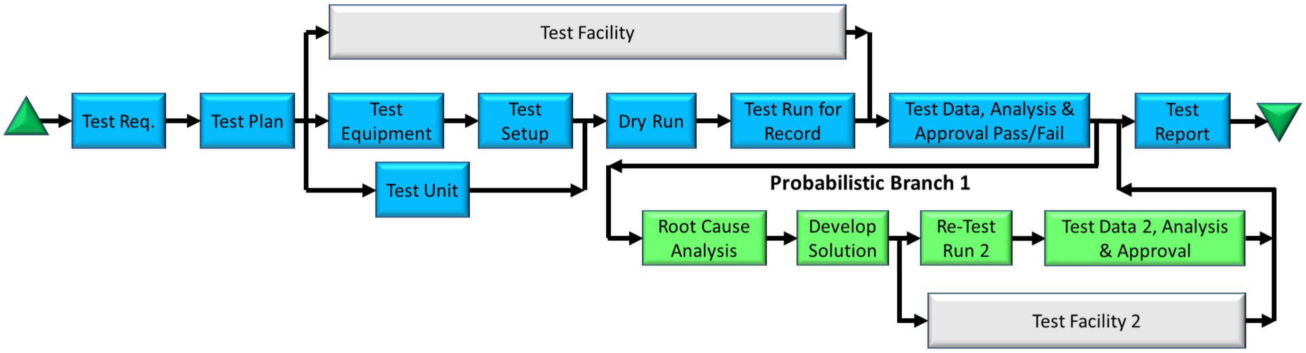


**Figure 4. Case 2 Added probabilistic branch to model risk**

as planned, they only extend (or change) current planned duration within a specified range. No new tasks, that may be needed, are added.

A more representative method of analysis is to include branching alternatives if tasks do not go as planned. The benefit of branching is in the fourth method (simulation). It automates the "what-if" modeling shown in the first three methods. In addition, the simulations provide a probability, or confidence level based on the branching likelihood with the resulting impact.

## 4. EXAMPLES

To illustrate the sensitivity of different modeling assumptions and resulting outcomes, we initial use a small project. In this example the test of a sub-system (test unit) is required. An outside test facility that has specialized equipment and tools to evaluate the unit will be used. The test requirements will be finalized as part of the project. Test equipment set up and a dry run will be conducted. The unit will be shipped to the facility. Following arrival of the unit the test will be run for record. When complete, test data will be reviewed along with any analysis to support a test approval decision. The last task is to publish a test report. The project detailed tasks are shown in Figure 1. Refer to Figure 17 in the appendix for the detailed schedule showing the tasks and location of the probabilistic branches.

The next sub section shows the sensitivity of using the qualitative rule-based method versus more complex quantitative simulations to obtain more realistic impacts which are compared in a later section.

Figure 3 shows the baseline Case 1, a success driven schedule, that assumes test success on the first run. This simulation models duration uncertainty factoring in the rule-based percentages for the identified risk. Case 2 presented in Figure 4 includes a model with a possible test failure and a probabilistic branch for diagnosis, repair and retest. For this simulation the branch is set with a 50% probability of occurrence (from a SME evaluation). Case 3 presented in Figure 5 is slightly more complex. It includes a second (nested) probabilistic branch that encompasses a redesign step if the test failure was due to a design flaw. For this simulation, both branches are set with a 50% probability of occurrence. In each case the impact to schedule and cost will be compared to the traditional rule-based method.

### *Traditional Rule-Based Method*

First, a risk of test failure has been identified and has been rated as medium. Using the criteria presented in Table 1 and

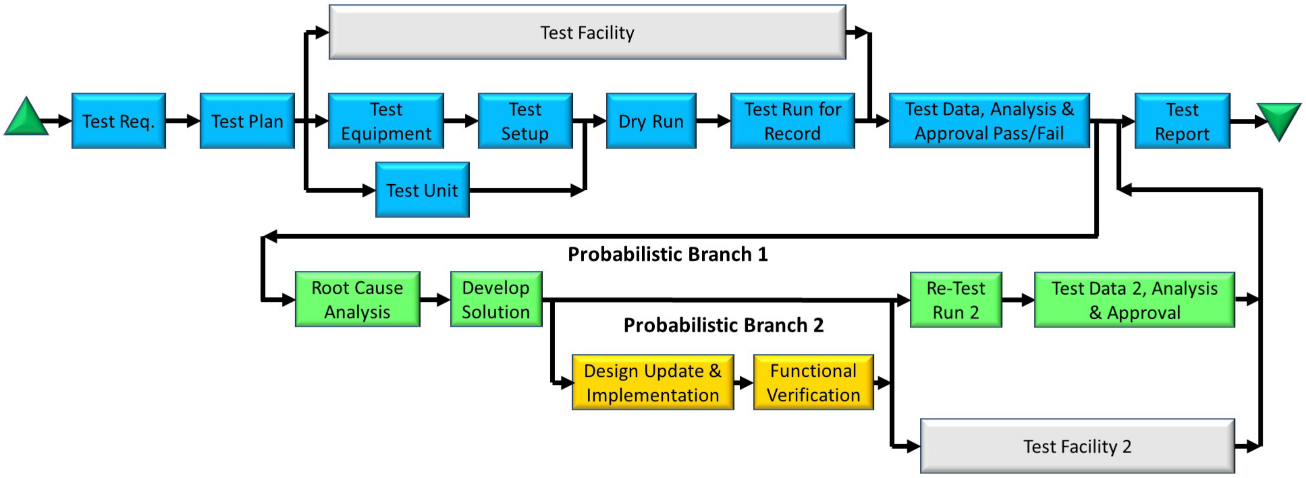


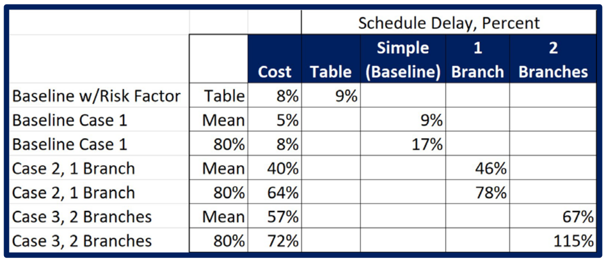
**Figure 5. Case 3 Addition of a second probabilistic branch to model possible design issue**

plotted in the 5 X 5 matrix of Figure 2. The cost and schedule impact are both 5% - 10%. The Baseline project duration is 46 days and costs are estimated at $164,300.

Using the qualitative risk guidance from Table 1 there is a 50% chance of a test failure resulting in a 5% - 10% impact on schedule slippage and cost growth. Quantifying, if it occurs, the test duration will be 48 - 51 days and cost will be $172,515 - $180,730. Activities within the failure period will include a root cause analysis of the failure, development of a solution, repair and retest.

The qualitative rule-based method indicates the likely schedule completion with a 10% duration delay changes to 9/2/2022 from 8/29/2022; and cost reserves are approximately 7.5% or about $12,000. This method provides a first order qualitative assessment and impact range based on judgement with no ability to provide a level of confidence the impact will be. In the absence of any additional information, this risk will be monitored with schedule margin and cost reserves set aside as needed.

**Table 3. Case Comparison Results**



| | | | Schedule Delay, Percent | | | |
|---|---|---|---|---|---|---|
| | | Cost | Table | Simple (Baseline) | 1 Branch | 2 Branches |
| Baseline w/Risk Factor | Table | 8% | 9% | | | |
| Baseline Case 1 | Mean | 5% | | 9% | | |
| Baseline Case 1 | 80% | 8% | | 17% | | |
| Case 2, 1 Branch | Mean | 40% | | | 46% | |
| Case 2, 1 Branch | 80% | 64% | | | 78% | |
| Case 3, 2 Branches | Mean | 57% | | | | 67% |
| Case 3, 2 Branches | 80% | 72% | | | | 115% |

*Simple Schedule Example (Case 1)*

Similar to the rule-based method, a probabilistic simulation was performed on the success driven schedule of Figure 3 with general uncertainty of pessimistic durations of 30% and optimistic durations of 10% (forming triangular distributions of 90%, 100% and 130%). The results provide a favorable comparison with the rule-based method and are consistent with a completion date of 9/2/2022 and cost reserves of $9K at the mean and $14K at an 80% confidence level. It infers that if there was a test failure, the likely recovery would be minimal impact based on the medium risk assignment. The benefit of this method over the traditional method is that it provides a confidence level to meet a date and likely cost if the risk occurs. Note however, this is a success driven schedule. This simple model does not necessarily address all additional activities needed (change on course of action) to diagnosis the test failure such as root cause, finding a solution, repair it and then retesting. It only assumes a longer duration in existing tasks.

*Single Branch Example (Case 2)*

A more representative model would include at least one probabilistic branch to address the likely activities needed to resolve the test failure and retest the unit. This is presented in Figure 4. The branch is at the end of the first test and the approval (Pass/Fail) decision point. In this case, the probability of including the branch is driven by the possibility of having a test failure. In the example a 50% probability is used, for the medium risk identified in the traditional method with no additional information available. If a test failure occurred, the activities would include a root cause analysis, development and implementation of a solution; then re-run the test. This model more realistically supports how the project might be done. It captures the additional resources and duration needed. The results in this case now indicate 46% - 78% schedule growth. That is close to a three week to one month slip in completion and a 40% – 64% cost growth. This is significantly larger than the traditional rule-based method or the simple probabilistic model (Case 1).

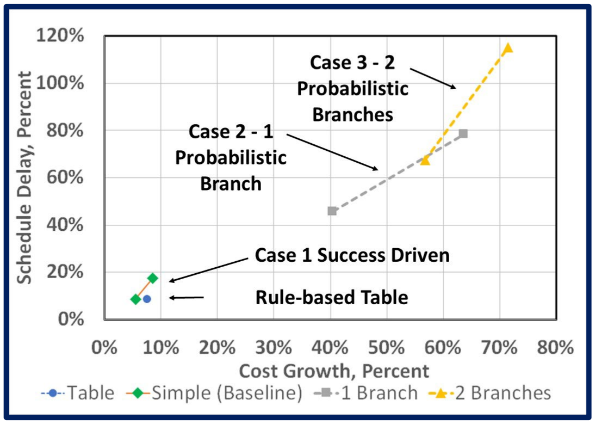


**Figure 6. Comparison results of each model**

*Double Branch Example (Case 3)*

The most representative model would have an additional branch if a test failure occurred. This second branch would be nested within the first branch and would include a redesign and implementation (again, modeled at a 50% probability) if the root cause revealed a design or workmanship issue. This is illustrated in Figure 5 following the root cause analysis. Based on this model the results indicate 67% - 115% schedule growth or a five to seven weeks delay and a 57% - 72% cost growth. This model is the most comprehensive and provides more information to make decisions. If performed early in the planning process, mitigation steps to reduce or avoid the risk could be performed, reducing or even avoiding the probability of occurrence with minimal disruption.

*Summary of the Examples*

When comparing the methods (qualitative traditional rule-based method and quantitative probabilistic simulation method) and using success driven models versus higher fidelity more representative models, simulations provide significantly more information to make decisions. The results presented in Table 2 show the significant differences driving program managers, system engineers and other stakeholders to perform higher fidelity modeling when objectively evaluating project risks. Figure 6 illustrates the significant differences. For the traditional rule-based method and Case 1 success driven uncertainty analysis are very similar and would generally be accepted as adequate. However, looking a little deeper, a single probabilistic branch significantly changes the results and with a little more fidelity using two probabilistic branches, the results provide decisionmakers significant value at minimal cost. It would allow time and resources to mitigate or avoid the risk and have adequate quantifiable (schedule and cost) reserves if a test failure occurs.

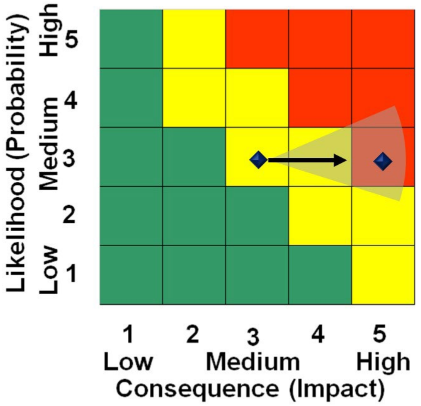


Figure 7. Risk adjusted based on simulation results

Reviewing the results, if the rule-based method is used, the risk should have been rated high versus medium based on the simulation results. Since the schedule delay and cost growth are likely greater than 20%, the simulations add value for stakeholders. Figure 7 shows how the risk might be adjusted in the risk cube. Still assuming a 50% probability of occurrence.

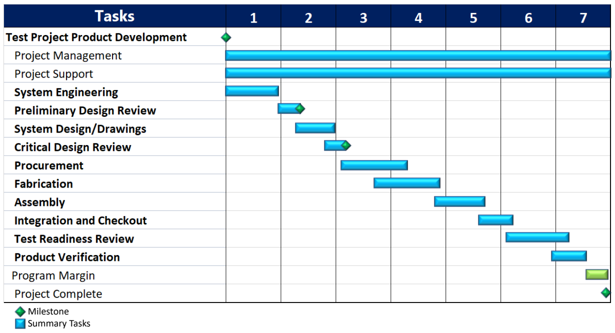


Figure 8. Project Summary Schedule Overview

## 5. PROJECT VIEW

When adding multiple risks and branching in a project, the simulations provide a "portfolio" view of the impacts and allow for appropriate cost and time reserves to meet expectations.

The project for this example contains eleven summary tasks ranging from level of effort project management activity to discrete tasks including system engineering, design, fabrication quality, product verification, etc. This is illustrated in Figure 8. The lower-level tasks (shown in Appendix C) contain both discrete risks and branching risks for the test program.

Table 4. List of discrete risks modeled in the project

| Risk ID | Probability | Duration | Cost Impact | Description | Tasks IDs Impacted |
|---|---|---|---|---|---|
| Proj1Rsk1 | 50% | 15d | $0 | Requirements uncertainty, taking longer than planned to complete | 6 |
| Proj1Rsk2 | 50% | 30d | $0 | Drawing completion taking longer than planned. | 16, 18 |
| Proj1Rsk3 | 30% | 30d | $20,000 | Mechanical Procurement taking longer than planned and increased cost. | 17, 24 |
| Proj1Rsk4 | 30% | 30d | $50,000 | Electrical components take longer than planned and cost more. | 25 |
| Proj1Rsk5 | 50% | 40d | $25,000 | Mechanical fabrication taking longer than planned and added costs. | 29 |
| Proj1Rsk6 | 60% | 0d | $50,000 | Purchased Parts cost more. | 30, 31 |
| Proj1Rsk7 | 50% | 45d | $15,000 | Mech/Elect integration taking longer than planned. | 41 |
| Proj1Rsk8 | 80% | 60d | $25,000 | Additional effort for finishing software. | 42 |

The simulation contains eight discrete risks identified for the program ranging from requirements stability, effort taking longer than planned, parts cost inflation and additional effort for software development. Each of the risks are associated with specific tasks in the schedule. Probabilities and schedule and or cost impacts are also defined. When running simulations, if there is no specific cost impact with a duration, the estimated cost impact is developed with differences in duration within the model. Table 4 is a summary of the discrete risks for both cost and schedule.

After running simulations estimating time and cost reserves can be estimated more reliably. Nominal generally accepted practices are typically factored cost and schedule reserves

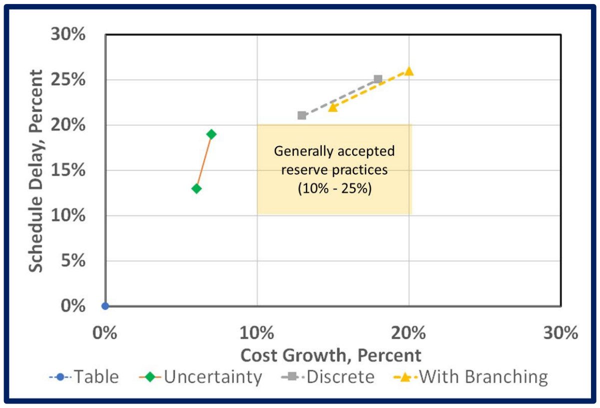


Figure 9. Simulation summary vs generally accepted reserve practices.

ranging from 10% - 20%. When simulations are integrated into the planning mix, it becomes clear that the generally accepted practices may not be valid.

When evaluating the needed reserves, if schedule (and cost) uncertainty is evaluated and no further analysis is done, it appears the generally accepted practices would be adequate shown by the green curve of Figure 9. However, when discrete risks and branching are modeled, that more realistically evaluate program execution, we see that the generally accepted reserves may not be adequate.

**Table 5. Uncertainty vs simulation sensitivities.**

| | | Schedule Delay, Percent | | | |
|---|---|---|---|---|---|
| | **Cost** | **Table** | **Uncertainty** | **Discrete** | **With Branching** |
| Baseline | 0% | 0% | | | |
| Mean | 6% | | 13% | | |
| 80% | 7% | | 19% | | |
| Mean | 13% | | | 21% | |
| 80% | 18% | | | 25% | |
| Mean | 15% | | | | 22% |
| 80% | 20% | | | | 26% |

When performing project planning activities modeling realistic program approaches can provide insight into cost and schedule drivers, changes in critical paths and defining realistic schedule and cost reserves beyond those over generally accepted practices. Table 5 is a summary of the forecast cost and schedule growth with uncertainty and risk simulations.

## 6. BENEFITS

*Traditional Approach*

The traditional rule-based or generally accepted practices is a qualitative method, and a starting point for identifying project or program risks and their respective ratings. It would be appropriate early in program definition. It is important to have some idea of the activities and product outcome if unfavorable events occur. These can then be collected and factored into the uncertainty analysis for both cost and schedule. The result can provide stakeholders some visibility into focus areas. When factoring these risks into the schedule, there is added visibility into where and what the impacts might be rather than performing a risk assessment without integrating the schedule activities.

*Simulations*

Adding representative simulations when evaluating identified risks add significant value to program visibility as demonstrated in both sets of examples. It can support a taxonomy update in the table structure if there are trends beyond those in Table 1 (or organization specific values). With the minor additional effort to build a representative simulation, the value to support early mitigation or avoidance can be budgeted and executed with little impact to the project as any confidence level.

*Integrating a Quantitative Risk Gate*

Rule-based risk definition is a starting point in the process. Setting up guidelines for risks that can significantly impact a programs outcome should implement additional rigor in the risk definition and quantification. A recommended approach would be to define a maximum threshold of impacts (cost and schedule) and probabilities high impact risks to develop higher fidelity analysis like the sample shown in this paper. This gated approach will provide stakeholders and managers insights to nascent pitfalls to address during program execution that might otherwise be overlooked.

## 7. SUMMARY

During project execution, risk management is integral to the process. Starting with a qualitative review provides a first order assessment that can help identify impacts and outcomes to budget, time and cost reserves. This activity provides visibility into where to look for trouble areas. However, when using only rule-based methods, the impacts are generally limited to ranges identified in the organization's probability and impact tables. Furthermore, schedule impacts are most often converted to cost or by simply adding time to existing tasks without regard to adding a set of likely tasks to correct anomalies (if) when they occur. Moreover, when schedule impacts are translated to cost without regard for impacts on time, the existing "success" driven schedule forecast will appear to consider true risks but provide an optimistic solution that may not represent actual likely performance.

As shown, using the simulation method, the next level of fidelity provides objectively to realistically evaluate risks for representative scenarios beyond the rule-based method and provide a confidence driven solution with specific time impacts at defined confidence levels. Appendix B illustrates the impacts in Figure 13 and Figure 15.

Quantitative simulations considered early in program execution when more data is available can provide significantly more information for addressing risk(s). They support actionable activities within programs and portfolios, sometimes with many risks, that add realism and implement early mitigations to reduce impacts and delays if the anomalous events occur. Integrating the risk process into the program schedule with branching supports stakeholder decision making beyond the qualitative assessment and assures mitigation steps are timely to minimize project impacts.

## 8. FUTURE WORK

Future research will include evaluating historical scenarios across a myriad of risk types and applying advanced methods such as Machine Learning and Artificial Intelligence to broaden trade space and reduce time to obtain recommendations. In addition, having tools to quickly make recommendations support schedule and budget planning for early mitigations that meet critical program delivery commitments and integrate adequate schedule and cost reserves that support program execution success.

## REFERENCES

[1] GAO-22-105230 "Weapon Systems Annual Assessment, June 2022.
[2] US Airforce Cost Risk and Uncertainty Handbook, April 2007, Pg. 1.
[3] Government Accountability Office GAO-16-89G, "Schedule Assessment Guide: Best Practices for Project Schedules", Pg 100.
[4] Office of the Deputy Assistant Secretary of Defense for System Engineering, "Department of Defense Risk Management Guide for Defense Acquisition Programs", 7th Edition, 2014
[5] Hoffman, E. and Lawbaugh, W., "Issues in NASA Program and Project Management – A collection of papers on NASA Procedures and Guidance 7120.5A, NASA/SP-98-6101/Vol14, 1998.
[6] NASA/SP-2011-3422, "NASA Risk Management Handbook, 2011.
[7] Graves, R. (2000). Qualitative risk assessment. PM Network, 14(10), 61–66.
[8] Kaplan, R. and Mikes, A. "Managing Risks: A New Framework", Harvard Business Review, June 2012.
[9] AACE International Recommended Practice 32R-04 "Determining Activity Durations", 2012, Pg. 8.
[10] Hulett, D. and Druker, E. "Integrating with Project Schedule Risk Improves Analysis of Cost Risk, AACEI Technical Paper RISK.1698, 2014.

## BIOGRAPHY

***Patrick Malone** is a principal at Systems Planning and Analysis, Inc. He has analytical and hands-on experience in the aerospace and space industries. He has written numerous papers on technology, cost estimating and scheduling. He has a BS from Arizona State and an MBA from Pepperdine University. He is a ICEAA CCE/A.*

# APPENDICES

## APPENDIX A

GAO Knowledge Point Framework for Program Maturity

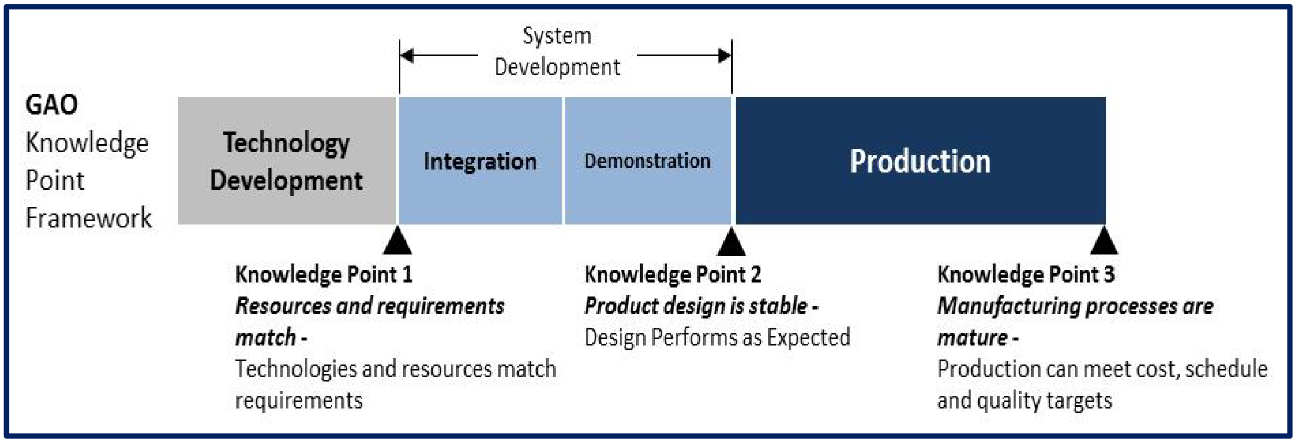


**Figure 10. GAO Knowledge Point Framework vs Program Lifecycle.**

**Knowledge point 1**: Resources and requirements match. Achieving a high level of technology maturity by the start of system development is one of several important indicators of whether this match has been made. This means that the technologies needed to meet essential product requirements have been demonstrated to work in a relevant environment. In addition, the developer should complete a series of systems engineering reviews culminating in a preliminary design of the product that shows the design is feasible. Constraining the development phase of a program to 5 or 6 years is also recommended because it aligns with DOD's budget planning process and fosters the negotiation of trade-offs in requirements and technologies. For shipbuilding programs, critical technologies should be matured into actual sub-system prototypes and successfully demonstrated in an operational environment before a contract is awarded for the detailed design of a new ship.

**Knowledge point 2**: Product design is stable. This point occurs when a program determines that a product's design will meet customer requirements, as well as cost, schedule, and reliability targets. A best practice is to achieve design stability at the system-level critical design review, usually held midway through system development. Completion of at least 90 percent of engineering drawings at this point provides tangible evidence that the product's design is stable, and a prototype demonstration shows that the design is capable of meeting performance requirements. Shipbuilding programs should demonstrate design stability by completing 100 percent of the basic and functional drawings, as well as the three-dimensional product model by the start of construction for a new ship. Programs can also improve the stability of their design by conducting reliability growth testing and completing failure modes and effects analyses so fixes can be incorporated before production begins. At this point, programs should also begin preparing for production by identifying manufacturing risks, key product characteristics, and critical manufacturing processes.

**Knowledge point 3**: Manufacturing processes are mature. This point is achieved when it has been demonstrated that the developer can manufacture the product within cost, schedule, and quality targets. A best practice is to ensure that all critical manufacturing processes are in statistical control—that is, they are repeatable, sustainable, and capable of consistently producing parts within the product's quality tolerances and standards—at the start of production. Demonstrating critical processes on a pilot production line is an important initial step in this effort. In addition, production and postproduction costs are minimized when a fully integrated, capable production-representative prototype is demonstrated to show that the system will work as intended in a reliable manner before committing to production. We did not assess shipbuilding programs for this knowledge point due to differences in the production processes used to build ships.

# APPENDIX B

Example Results

The following figures provide the analytical details of the simulations for Cases 1, 2 & 3. The figures are in two groups for each case. The first group presents the probability distributions, confidence for end date schedule in the top image and likely cost in the bottom image. The second group provides the criticality index in the top image and the task drivers in the bottom image.

Figure 8 is Case 1 the success driven simulation. It provides a classical "S" curve showing date and cost based on confidence level. Figure 9 presents the criticality index and the time tasks are on the critical path. The bottom image is the task driver tornado chart showing tasks driving the completion date.

Figure 10 is Case 2 with a single branch. It shows a bimodal distribution based on diagnosis, resolution and retest with the corresponding cost distribution. Figure 11 presents the criticality index and the time tasks are on the critical path. The bottom image is the task driver tornado chart showing tasks driving the completion date.

Figure 12 is Case 3 a double branch. It shows a tri-modal distribution based on a diagnosis, redesign, implementation and retest. Figure 13 presents the criticality index and the time tasks are on the critical path. The bottom image is the task driver tornado chart showing tasks driving the completion date.

Figure 14 is the project schedule showing the location of the probabilistic branches. Case 1 is without the branches; Case 2 implements branch 1 and Case 3 implements branches 1 & 2.

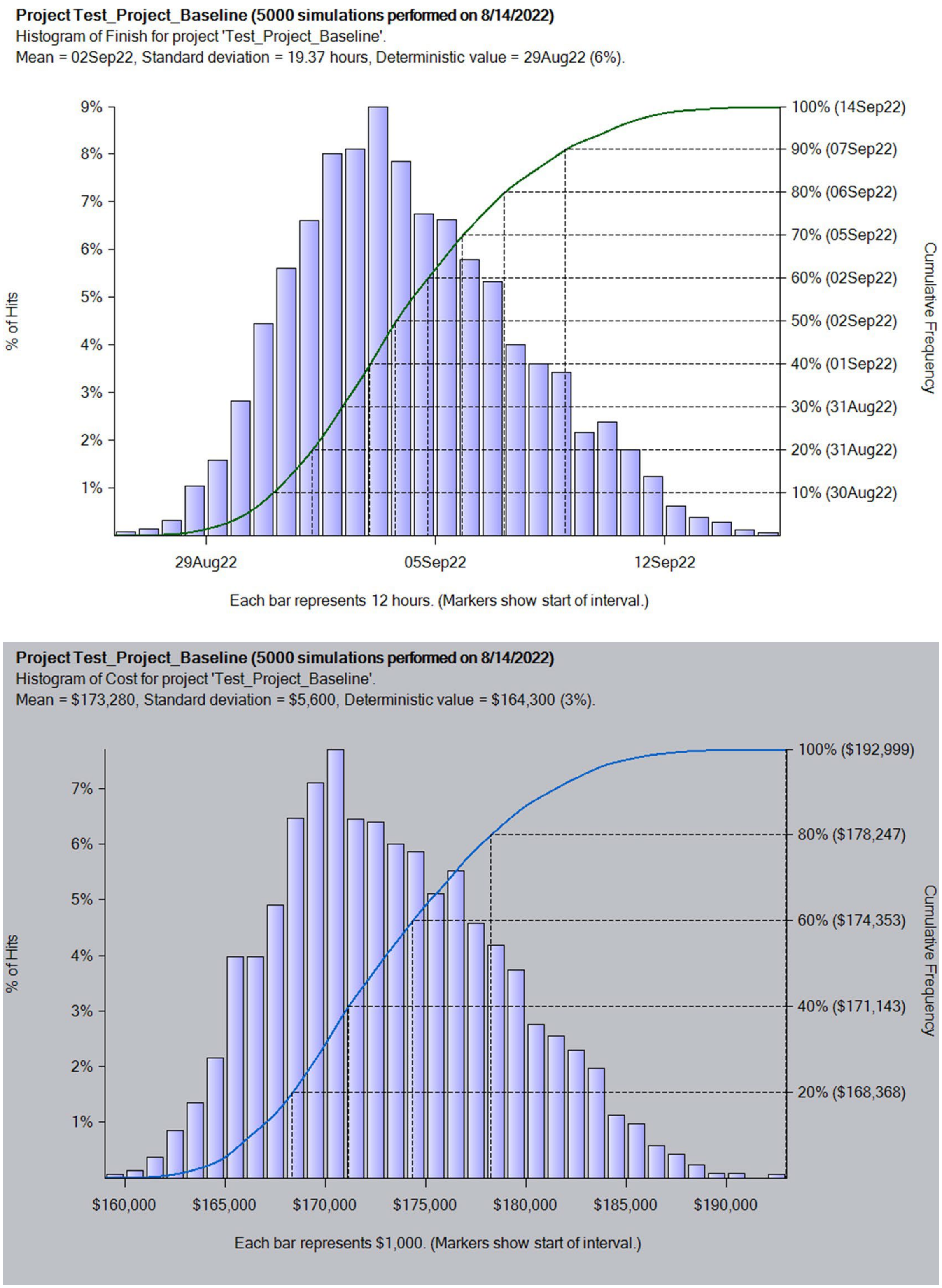
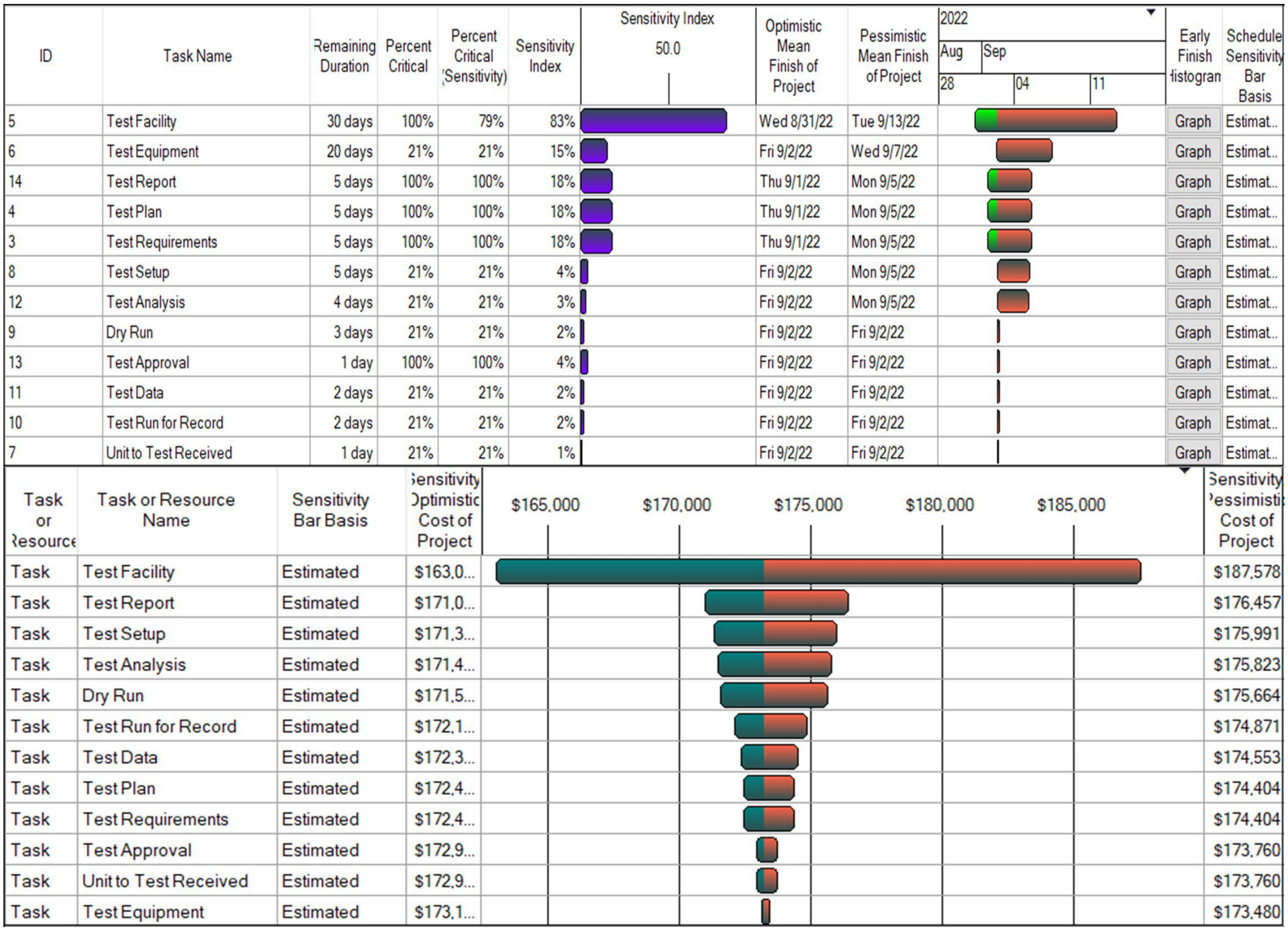


**Figure 11 – Case 1 Probability Distributions for Schedule and Cost**



| ID | Task Name | Remaining Duration | Percent Critical | Percent Critical (Sensitivity) | Sensitivity Index | Sensitivity Index 50.0 | Optimistic Mean Finish of Project | Pessimistic Mean Finish of Project | 2022 Aug Sep 28 04 11 | Early Finish Histogram | Schedule Sensitivity Bar Basis |
|---|---|---|---|---|---|---|---|---|---|---|---|
| 5 | Test Facility | 30 days | 100% | 79% | 83% | | Wed 8/31/22 | Tue 9/13/22 | | Graph | Estimat... |
| 6 | Test Equipment | 20 days | 21% | 21% | 15% | | Fri 9/2/22 | Wed 9/7/22 | | Graph | Estimat... |
| 14 | Test Report | 5 days | 100% | 100% | 18% | | Thu 9/1/22 | Mon 9/5/22 | | Graph | Estimat... |
| 4 | Test Plan | 5 days | 100% | 100% | 18% | | Thu 9/1/22 | Mon 9/5/22 | | Graph | Estimat... |
| 3 | Test Requirements | 5 days | 100% | 100% | 18% | | Thu 9/1/22 | Mon 9/5/22 | | Graph | Estimat... |
| 8 | Test Setup | 5 days | 21% | 21% | 4% | | Fri 9/2/22 | Mon 9/5/22 | | Graph | Estimat... |
| 12 | Test Analysis | 4 days | 21% | 21% | 3% | | Fri 9/2/22 | Mon 9/5/22 | | Graph | Estimat... |
| 9 | Dry Run | 3 days | 21% | 21% | 2% | | Fri 9/2/22 | Fri 9/2/22 | | Graph | Estimat... |
| 13 | Test Approval | 1 day | 100% | 100% | 4% | | Fri 9/2/22 | Fri 9/2/22 | | Graph | Estimat... |
| 11 | Test Data | 2 days | 21% | 21% | 2% | | Fri 9/2/22 | Fri 9/2/22 | | Graph | Estimat... |
| 10 | Test Run for Record | 2 days | 21% | 21% | 2% | | Fri 9/2/22 | Fri 9/2/22 | | Graph | Estimat... |
| 7 | Unit to Test Received | 1 day | 21% | 21% | 1% | | Fri 9/2/22 | Fri 9/2/22 | | Graph | Estimat... |

| Task or Resource | Task or Resource Name | Sensitivity Bar Basis | Sensitivity Optimistic Cost of Project | $165,000 $170,000 $175,000 $180,000 $185,000 | Sensitivity Pessimistic Cost of Project |
|---|---|---|---|---|---|
| Task | Test Facility | Estimated | $163,0... | | $187,578 |
| Task | Test Report | Estimated | $171,0... | | $176,457 |
| Task | Test Setup | Estimated | $171,3... | | $175,991 |
| Task | Test Analysis | Estimated | $171,4... | | $175,823 |
| Task | Dry Run | Estimated | $171,5... | | $175,664 |
| Task | Test Run for Record | Estimated | $172,1... | | $174,871 |
| Task | Test Data | Estimated | $172,3... | | $174,553 |
| Task | Test Plan | Estimated | $172,4... | | $174,404 |
| Task | Test Requirements | Estimated | $172,4... | | $174,404 |
| Task | Test Approval | Estimated | $172,9... | | $173,760 |
| Task | Unit to Test Received | Estimated | $172,9... | | $173,760 |
| Task | Test Equipment | Estimated | $173,1... | | $173,480 |

**Figure 12 – Case 1 Critical Indices and Task drivers**

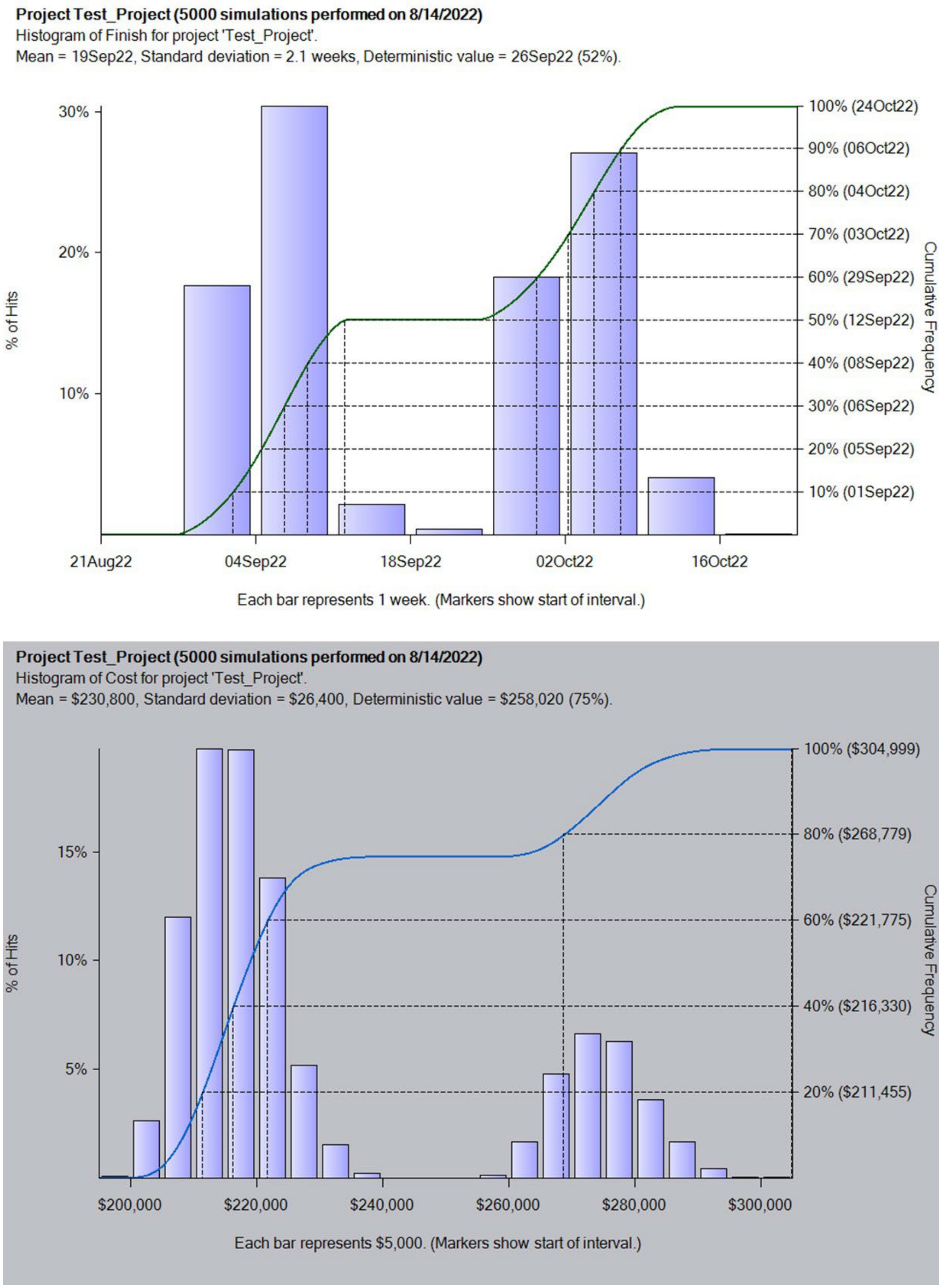
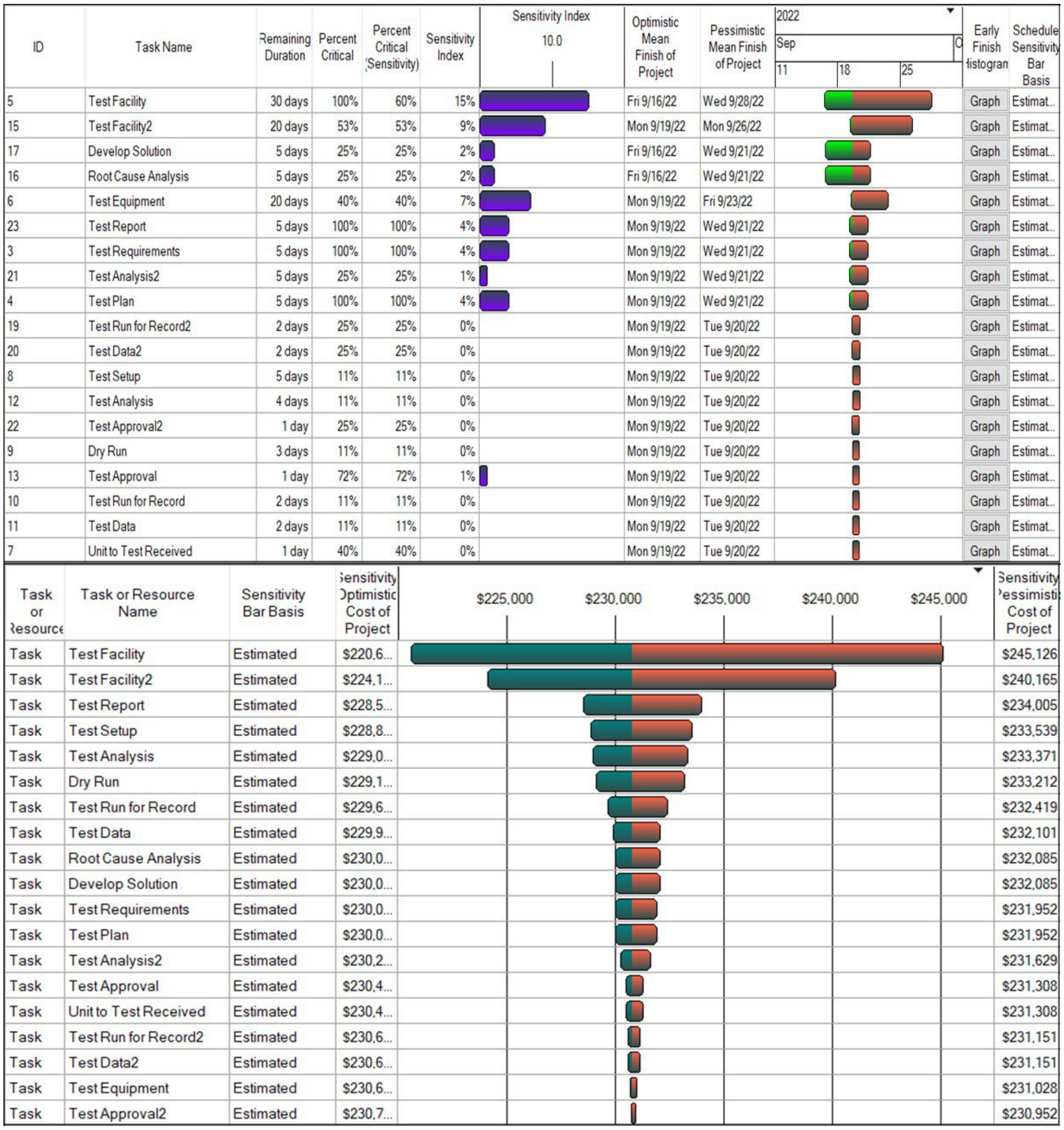


**Figure 13 – Case 2 Probability distribution for schedule and cost**



| ID | Task Name | Remaining Duration | Percent Critical | Percent Critical (Sensitivity) | Sensitivity Index | Optimistic Mean Finish of Project | Pessimistic Mean Finish of Project | Early Finish istogran | Schedule Sensitivity Bar Basis |
|---|---|---|---|---|---|---|---|---|---|
| 5 | Test Facility | 30 days | 100% | 60% | 15% | Fri 9/16/22 | Wed 9/28/22 | Graph | Estimat... |
| 15 | Test Facility2 | 20 days | 53% | 53% | 9% | Mon 9/19/22 | Mon 9/26/22 | Graph | Estimat... |
| 17 | Develop Solution | 5 days | 25% | 25% | 2% | Fri 9/16/22 | Wed 9/21/22 | Graph | Estimat... |
| 16 | Root Cause Analysis | 5 days | 25% | 25% | 2% | Fri 9/16/22 | Wed 9/21/22 | Graph | Estimat... |
| 6 | Test Equipment | 20 days | 40% | 40% | 7% | Mon 9/19/22 | Fri 9/23/22 | Graph | Estimat... |
| 23 | Test Report | 5 days | 100% | 100% | 4% | Mon 9/19/22 | Wed 9/21/22 | Graph | Estimat... |
| 3 | Test Requirements | 5 days | 100% | 100% | 4% | Mon 9/19/22 | Wed 9/21/22 | Graph | Estimat... |
| 21 | Test Analysis2 | 5 days | 25% | 25% | 1% | Mon 9/19/22 | Wed 9/21/22 | Graph | Estimat... |
| 4 | Test Plan | 5 days | 100% | 100% | 4% | Mon 9/19/22 | Wed 9/21/22 | Graph | Estimat... |
| 19 | Test Run for Record2 | 2 days | 25% | 25% | 0% | Mon 9/19/22 | Tue 9/20/22 | Graph | Estimat... |
| 20 | Test Data2 | 2 days | 25% | 25% | 0% | Mon 9/19/22 | Tue 9/20/22 | Graph | Estimat... |
| 8 | Test Setup | 5 days | 11% | 11% | 0% | Mon 9/19/22 | Tue 9/20/22 | Graph | Estimat... |
| 12 | Test Analysis | 4 days | 11% | 11% | 0% | Mon 9/19/22 | Tue 9/20/22 | Graph | Estimat... |
| 22 | Test Approval2 | 1 day | 25% | 25% | 0% | Mon 9/19/22 | Tue 9/20/22 | Graph | Estimat... |
| 9 | Dry Run | 3 days | 11% | 11% | 0% | Mon 9/19/22 | Tue 9/20/22 | Graph | Estimat... |
| 13 | Test Approval | 1 day | 72% | 72% | 1% | Mon 9/19/22 | Tue 9/20/22 | Graph | Estimat... |
| 10 | Test Run for Record | 2 days | 11% | 11% | 0% | Mon 9/19/22 | Tue 9/20/22 | Graph | Estimat... |
| 11 | Test Data | 2 days | 11% | 11% | 0% | Mon 9/19/22 | Tue 9/20/22 | Graph | Estimat... |
| 7 | Unit to Test Received | 1 day | 40% | 40% | 0% | Mon 9/19/22 | Tue 9/20/22 | Graph | Estimat... |

| Task or esource | Task or Resource Name | Sensitivity Bar Basis | ensitivity ptimistic Cost of Project | Sensitivity essimisti Cost of Project |
|---|---|---|---|---|
| Task | Test Facility | Estimated | $220,6... | $245,126 |
| Task | Test Facility2 | Estimated | $224,1... | $240,165 |
| Task | Test Report | Estimated | $228,5... | $234,005 |
| Task | Test Setup | Estimated | $228,8... | $233,539 |
| Task | Test Analysis | Estimated | $229,0... | $233,371 |
| Task | Dry Run | Estimated | $229,1... | $233,212 |
| Task | Test Run for Record | Estimated | $229,6... | $232,419 |
| Task | Test Data | Estimated | $229,9... | $232,101 |
| Task | Root Cause Analysis | Estimated | $230,0... | $232,085 |
| Task | Develop Solution | Estimated | $230,0... | $232,085 |
| Task | Test Requirements | Estimated | $230,0... | $231,952 |
| Task | Test Plan | Estimated | $230,0... | $231,952 |
| Task | Test Analysis2 | Estimated | $230,2... | $231,629 |
| Task | Test Approval | Estimated | $230,4... | $231,308 |
| Task | Unit to Test Received | Estimated | $230,4... | $231,308 |
| Task | Test Run for Record2 | Estimated | $230,6... | $231,151 |
| Task | Test Data2 | Estimated | $230,6... | $231,151 |
| Task | Test Equipment | Estimated | $230,6... | $231,028 |
| Task | Test Approval2 | Estimated | $230,7... | $230,952 |

Figure 14 - Case 2 Critical Indices and Task drivers

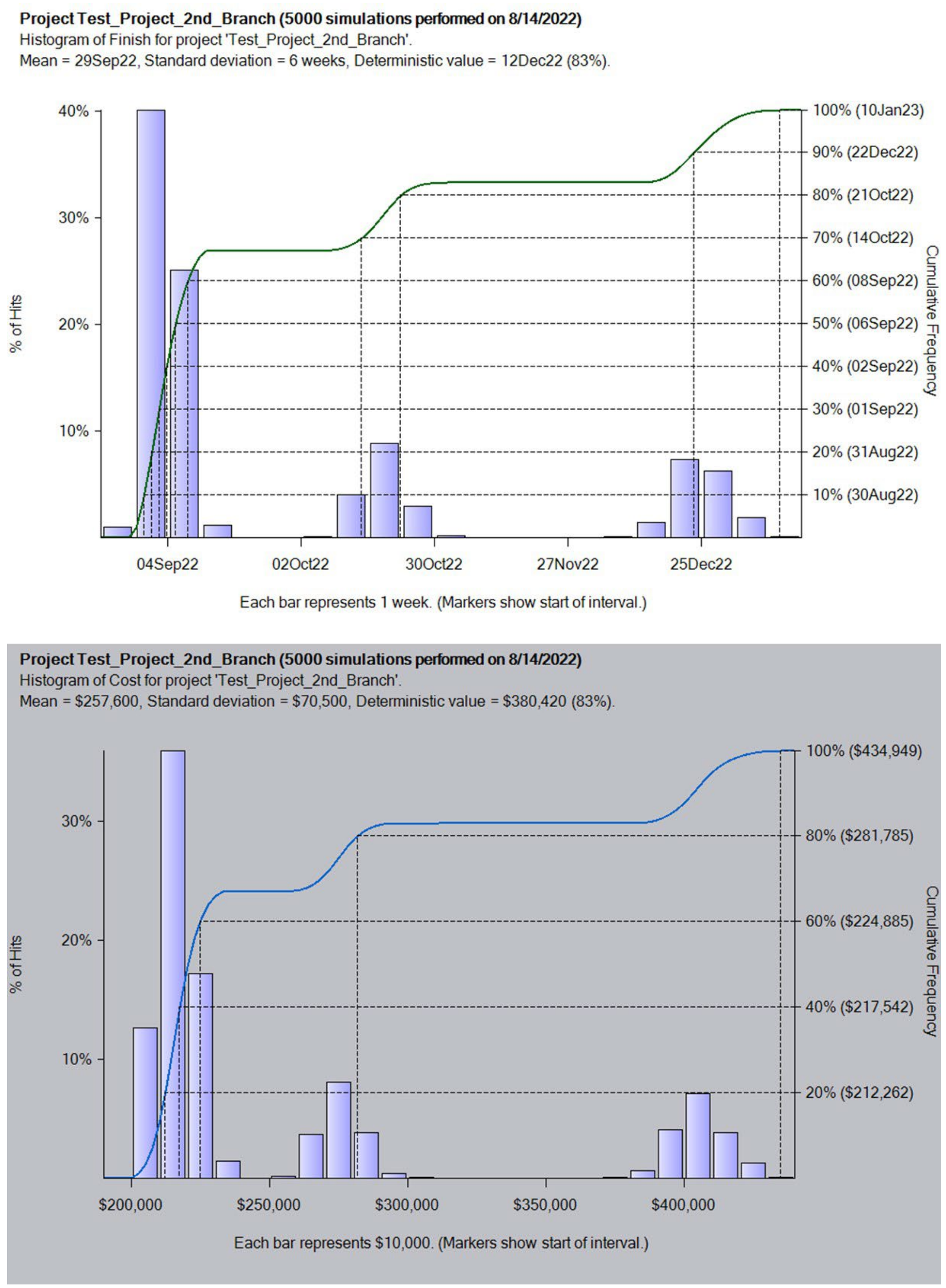
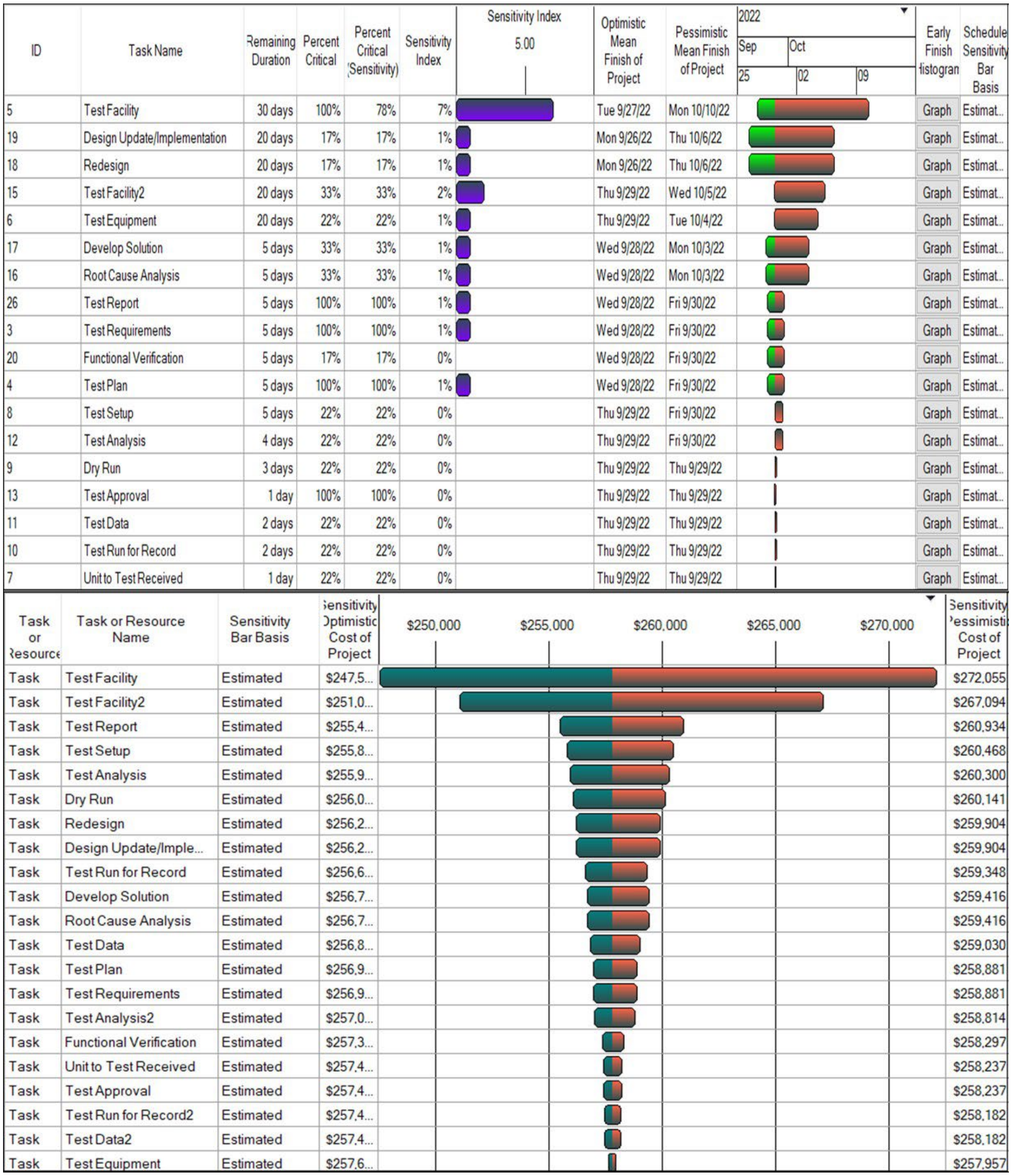


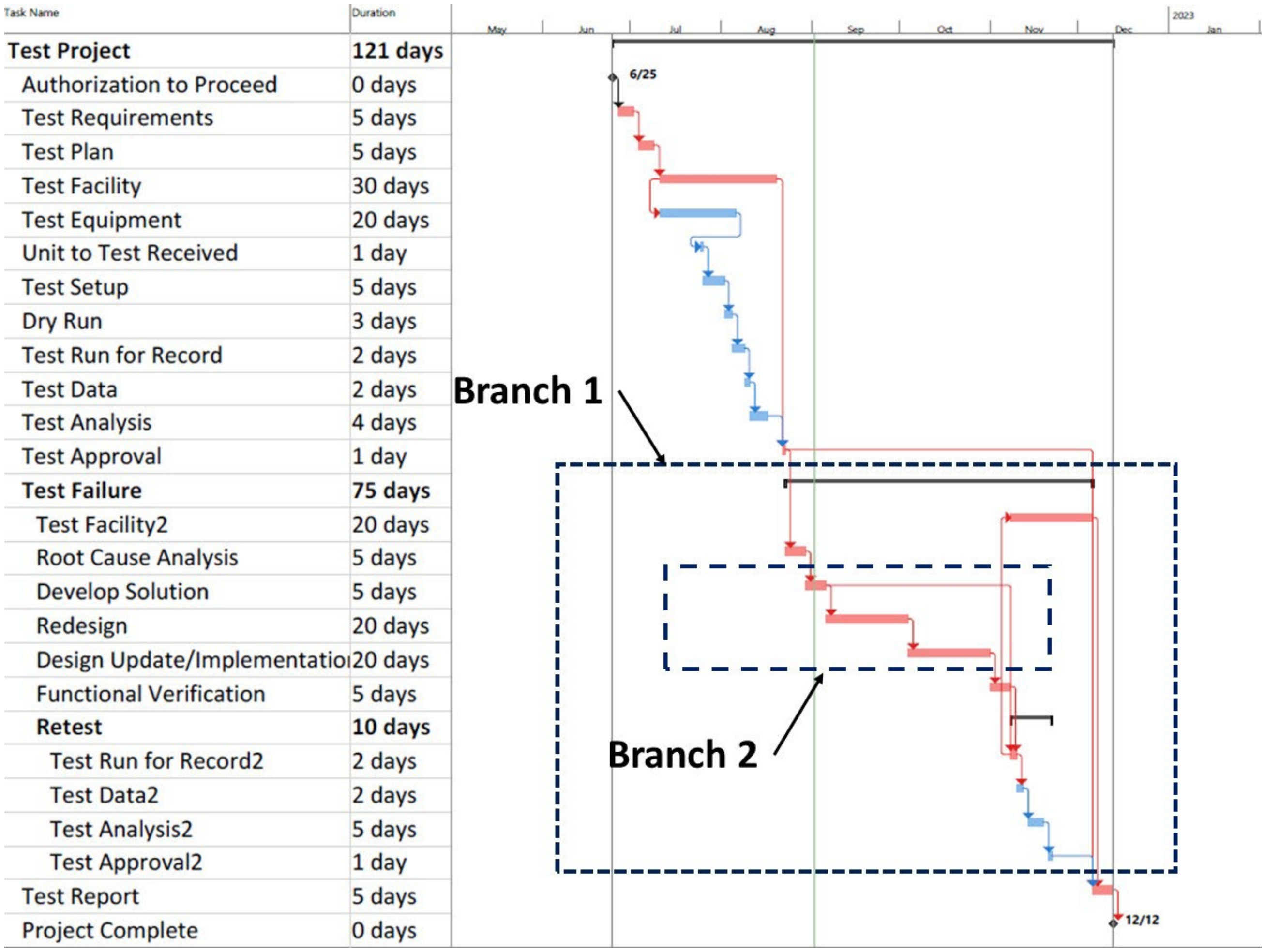
**Figure 15 - Case 3 Probability distribution for schedule and cost**



| ID | Task Name | Remaining Duration | Percent Critical | Percent Critical (Sensitivity) | Sensitivity Index | Sensitivity Index<br>5.00 | Optimistic Mean Finish of Project | Pessimistic Mean Finish of Project | 2022<br>Sep Oct<br>25 02 09 | Early Finish listogram | Schedule Sensitivity Bar Basis |
|---|---|---|---|---|---|---|---|---|---|---|---|
| 5 | Test Facility | 30 days | 100% | 78% | 7% | | Tue 9/27/22 | Mon 10/10/22 | | Graph | Estimat... |
| 19 | Design Update/Implementation | 20 days | 17% | 17% | 1% | | Mon 9/26/22 | Thu 10/6/22 | | Graph | Estimat... |
| 18 | Redesign | 20 days | 17% | 17% | 1% | | Mon 9/26/22 | Thu 10/6/22 | | Graph | Estimat... |
| 15 | Test Facility2 | 20 days | 33% | 33% | 2% | | Thu 9/29/22 | Wed 10/5/22 | | Graph | Estimat... |
| 6 | Test Equipment | 20 days | 22% | 22% | 1% | | Thu 9/29/22 | Tue 10/4/22 | | Graph | Estimat... |
| 17 | Develop Solution | 5 days | 33% | 33% | 1% | | Wed 9/28/22 | Mon 10/3/22 | | Graph | Estimat... |
| 16 | Root Cause Analysis | 5 days | 33% | 33% | 1% | | Wed 9/28/22 | Mon 10/3/22 | | Graph | Estimat... |
| 26 | Test Report | 5 days | 100% | 100% | 1% | | Wed 9/28/22 | Fri 9/30/22 | | Graph | Estimat... |
| 3 | Test Requirements | 5 days | 100% | 100% | 1% | | Wed 9/28/22 | Fri 9/30/22 | | Graph | Estimat... |
| 20 | Functional Verification | 5 days | 17% | 17% | 0% | | Wed 9/28/22 | Fri 9/30/22 | | Graph | Estimat... |
| 4 | Test Plan | 5 days | 100% | 100% | 1% | | Wed 9/28/22 | Fri 9/30/22 | | Graph | Estimat... |
| 8 | Test Setup | 5 days | 22% | 22% | 0% | | Thu 9/29/22 | Fri 9/30/22 | | Graph | Estimat... |
| 12 | Test Analysis | 4 days | 22% | 22% | 0% | | Thu 9/29/22 | Fri 9/30/22 | | Graph | Estimat... |
| 9 | Dry Run | 3 days | 22% | 22% | 0% | | Thu 9/29/22 | Thu 9/29/22 | | Graph | Estimat... |
| 13 | Test Approval | 1 day | 100% | 100% | 0% | | Thu 9/29/22 | Thu 9/29/22 | | Graph | Estimat... |
| 11 | Test Data | 2 days | 22% | 22% | 0% | | Thu 9/29/22 | Thu 9/29/22 | | Graph | Estimat... |
| 10 | Test Run for Record | 2 days | 22% | 22% | 0% | | Thu 9/29/22 | Thu 9/29/22 | | Graph | Estimat... |
| 7 | Unit to Test Received | 1 day | 22% | 22% | 0% | | Thu 9/29/22 | Thu 9/29/22 | | Graph | Estimat... |

| Task or Resource | Task or Resource Name | Sensitivity Bar Basis | Sensitivity Optimistic Cost of Project | $250,000 $255,000 $260,000 $265,000 $270,000 | Sensitivity Pessimisti Cost of Project |
|---|---|---|---|---|---|
| Task | Test Facility | Estimated | $247,5... | | $272,055 |
| Task | Test Facility2 | Estimated | $251,0... | | $267,094 |
| Task | Test Report | Estimated | $255,4... | | $260,934 |
| Task | Test Setup | Estimated | $255,8... | | $260,468 |
| Task | Test Analysis | Estimated | $255,9... | | $260,300 |
| Task | Dry Run | Estimated | $256,0... | | $260,141 |
| Task | Redesign | Estimated | $256,2... | | $259,904 |
| Task | Design Update/Imple... | Estimated | $256,2... | | $259,904 |
| Task | Test Run for Record | Estimated | $256,6... | | $259,348 |
| Task | Develop Solution | Estimated | $256,7... | | $259,416 |
| Task | Root Cause Analysis | Estimated | $256,7... | | $259,416 |
| Task | Test Data | Estimated | $256,8... | | $259,030 |
| Task | Test Plan | Estimated | $256,9... | | $258,881 |
| Task | Test Requirements | Estimated | $256,9... | | $258,881 |
| Task | Test Analysis2 | Estimated | $257,0... | | $258,814 |
| Task | Functional Verification | Estimated | $257,3... | | $258,297 |
| Task | Unit to Test Received | Estimated | $257,4... | | $258,237 |
| Task | Test Approval | Estimated | $257,4... | | $258,237 |
| Task | Test Run for Record2 | Estimated | $257,4... | | $258,182 |
| Task | Test Data2 | Estimated | $257,4... | | $258,182 |
| Task | Test Equipment | Estimated | $257,6... | | $257,957 |

**Figure 16 - Case 3 Critical Indices and Task drivers**

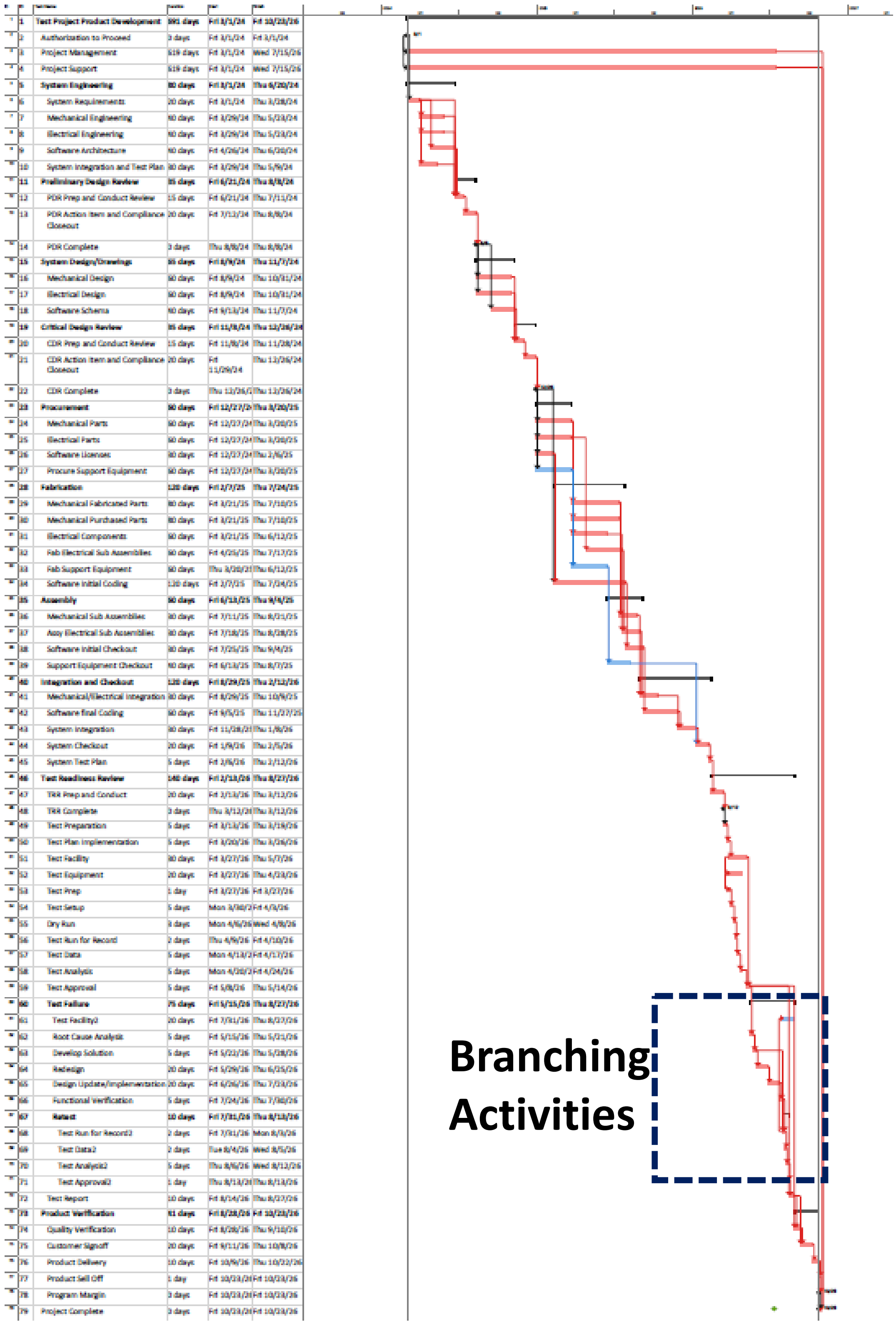


| Task Name | Duration |
|---|---|
| **Test Project** | **121 days** |
| Authorization to Proceed | 0 days |
| Test Requirements | 5 days |
| Test Plan | 5 days |
| Test Facility | 30 days |
| Test Equipment | 20 days |
| Unit to Test Received | 1 day |
| Test Setup | 5 days |
| Dry Run | 3 days |
| Test Run for Record | 2 days |
| Test Data | 2 days |
| Test Analysis | 4 days |
| Test Approval | 1 day |
| **Test Failure** | **75 days** |
| Test Facility2 | 20 days |
| Root Cause Analysis | 5 days |
| Develop Solution | 5 days |
| Redesign | 20 days |
| Design Update/Implementatio | 20 days |
| Functional Verification | 5 days |
| **Retest** | **10 days** |
| Test Run for Record2 | 2 days |
| Test Data2 | 2 days |
| Test Analysis2 | 5 days |
| Test Approval2 | 1 day |
| Test Report | 5 days |
| Project Complete | 0 days |

**Figure 17 – Project schedule used for analysis showing branch locations**

# APPENDIX C

## Project Summary



| [illegible] | [illegible] | [illegible] | [illegible] | [illegible] |
|---|---|---|---|---|
| 1 | **Test Project Product Development** | **691 days** | **Fri 3/1/24** | **Fri 10/23/26** |
| 2 | Authorization to Proceed | 0 days | Fri 3/1/24 | Fri 3/1/24 |
| 3 | Project Management | 619 days | Fri 3/1/24 | Wed 7/15/26 |
| 4 | Project Support | 619 days | Fri 3/1/24 | Wed 7/15/26 |
| 5 | **System Engineering** | **80 days** | **Fri 3/1/24** | **Thu 6/20/24** |
| 6 | System Requirements | 20 days | Fri 3/1/24 | Thu 3/28/24 |
| 7 | Mechanical Engineering | 40 days | Fri 3/29/24 | Thu 5/23/24 |
| 8 | Electrical Engineering | 40 days | Fri 3/29/24 | Thu 5/23/24 |
| 9 | Software Architecture | 40 days | Fri 4/26/24 | Thu 6/20/24 |
| 10 | System Integration and Test Plan | 30 days | Fri 3/29/24 | Thu 5/9/24 |
| 11 | **Preliminary Design Review** | **35 days** | **Fri 6/21/24** | **Thu 8/8/24** |
| 12 | PDR Prep and Conduct Review | 15 days | Fri 6/21/24 | Thu 7/11/24 |
| 13 | PDR Action Item and Compliance Closeout | 20 days | Fri 7/12/24 | Thu 8/8/24 |
| 14 | PDR Complete | 0 days | Thu 8/8/24 | Thu 8/8/24 |
| 15 | **System Design/Drawings** | **65 days** | **Fri 8/9/24** | **Thu 11/7/24** |
| 16 | Mechanical Design | 60 days | Fri 8/9/24 | Thu 10/31/24 |
| 17 | Electrical Design | 60 days | Fri 8/9/24 | Thu 10/31/24 |
| 18 | Software Schema | 40 days | Fri 9/13/24 | Thu 11/7/24 |
| 19 | **Critical Design Review** | **35 days** | **Fri 11/8/24** | **Thu 12/26/24** |
| 20 | CDR Prep and Conduct Review | 15 days | Fri 11/8/24 | Thu 11/28/24 |
| 21 | CDR Action Item and Compliance Closeout | 20 days | Fri 11/29/24 | Thu 12/26/24 |
| 22 | CDR Complete | 0 days | Thu 12/26/2 | Thu 12/26/24 |
| 23 | **Procurement** | **60 days** | **Fri 12/27/2** | **Thu 3/20/25** |
| 24 | Mechanical Parts | 60 days | Fri 12/27/24 | Thu 3/20/25 |
| 25 | Electrical Parts | 60 days | Fri 12/27/24 | Thu 3/20/25 |
| 26 | Software Licenses | 30 days | Fri 12/27/24 | Thu 2/6/25 |
| 27 | Procure Support Equipment | 60 days | Fri 12/27/24 | Thu 3/20/25 |
| 28 | **Fabrication** | **120 days** | **Fri 2/7/25** | **Thu 7/24/25** |
| 29 | Mechanical Fabricated Parts | 80 days | Fri 3/21/25 | Thu 7/10/25 |
| 30 | Mechanical Purchased Parts | 80 days | Fri 3/21/25 | Thu 7/10/25 |
| 31 | Electrical Components | 60 days | Fri 3/21/25 | Thu 6/12/25 |
| 32 | Fab Electrical Sub Assemblies | 60 days | Fri 4/25/25 | Thu 7/17/25 |
| 33 | Fab Support Equipment | 60 days | Thu 3/20/2 | Thu 6/12/25 |
| 34 | Software Initial Coding | 120 days | Fri 2/7/25 | Thu 7/24/25 |
| 35 | **Assembly** | **60 days** | **Fri 6/13/25** | **Thu 9/4/25** |
| 36 | Mechanical Sub Assemblies | 30 days | Fri 7/11/25 | Thu 8/21/25 |
| 37 | Assy Electrical Sub Assemblies | 30 days | Fri 7/18/25 | Thu 8/28/25 |
| 38 | Software Initial Checkout | 30 days | Fri 7/25/25 | Thu 9/4/25 |
| 39 | Support Equipment Checkout | 40 days | Fri 6/13/25 | Thu 8/7/25 |
| 40 | **Integration and Checkout** | **120 days** | **Fri 8/29/25** | **Thu 2/12/26** |
| 41 | Mechanical/Electrical Integration | 30 days | Fri 8/29/25 | Thu 10/9/25 |
| 42 | Software Final Coding | 60 days | Fri 9/5/25 | Thu 11/27/25 |
| 43 | System Integration | 30 days | Fri 11/28/2 | Thu 1/8/26 |
| 44 | System Checkout | 20 days | Fri 1/9/26 | Thu 2/5/26 |
| 45 | System Test Plan | 5 days | Fri 2/6/26 | Thu 2/12/26 |
| 46 | **Test Readiness Review** | **140 days** | **Fri 2/13/26** | **Thu 8/27/26** |
| 47 | TRR Prep and Conduct | 20 days | Fri 2/13/26 | Thu 3/12/26 |
| 48 | TRR Complete | 0 days | Thu 3/12/2 | Thu 3/12/26 |
| 49 | Test Preparation | 5 days | Fri 3/13/26 | Thu 3/19/26 |
| 50 | Test Plan Implementation | 5 days | Fri 3/20/26 | Thu 3/26/26 |
| 51 | Test Facility | 30 days | Fri 3/27/26 | Thu 5/7/26 |
| 52 | Test Equipment | 20 days | Fri 3/27/26 | Thu 4/23/26 |
| 53 | Test Prep | 1 day | Fri 3/27/26 | Fri 3/27/26 |
| 54 | Test Setup | 5 days | Mon 3/30/2 | Fri 4/3/26 |
| 55 | Dry Run | 3 days | Mon 4/6/26 | Wed 4/8/26 |
| 56 | Test Run for Record | 2 days | Thu 4/9/26 | Fri 4/10/26 |
| 57 | Test Data | 5 days | Mon 4/13/2 | Fri 4/17/26 |
| 58 | Test Analysis | 5 days | Mon 4/20/2 | Fri 4/24/26 |
| 59 | Test Approval | 5 days | Fri 5/8/26 | Thu 5/14/26 |
| 60 | **Test Failure** | **75 days** | **Fri 5/15/26** | **Thu 8/27/26** |
| 61 | Test Facility2 | 20 days | Fri 7/31/26 | Thu 8/27/26 |
| 62 | Root Cause Analysis | 5 days | Fri 5/15/26 | Thu 5/21/26 |
| 63 | Develop Solution | 5 days | Fri 5/22/26 | Thu 5/28/26 |
| 64 | Redesign | 20 days | Fri 5/29/26 | Thu 6/25/26 |
| 65 | Design Update/Implementation | 20 days | Fri 6/26/26 | Thu 7/23/26 |
| 66 | Functional Verification | 5 days | Fri 7/24/26 | Thu 7/30/26 |
| 67 | **Retest** | **10 days** | **Fri 7/31/26** | **Thu 8/13/26** |
| 68 | Test Run for Record2 | 2 days | Fri 7/31/26 | Mon 8/3/26 |
| 69 | Test Data2 | 2 days | Tue 8/4/26 | Wed 8/5/26 |
| 70 | Test Analysis2 | 5 days | Thu 8/6/26 | Wed 8/12/26 |
| 71 | Test Approval2 | 1 day | Thu 8/13/2 | Thu 8/13/26 |
| 72 | Test Report | 10 days | Fri 8/14/26 | Thu 8/27/26 |
| 73 | **Product Verification** | **41 days** | **Fri 8/28/26** | **Fri 10/23/26** |
| 74 | Quality Verification | 10 days | Fri 8/28/26 | Thu 9/10/26 |
| 75 | Customer Signoff | 20 days | Fri 9/11/26 | Thu 10/8/26 |
| 76 | Product Delivery | 10 days | Fri 10/9/26 | Thu 10/22/26 |
| 77 | Product Sell Off | 1 day | Fri 10/23/2 | Fri 10/23/26 |
| 78 | Program Margin | 0 days | Fri 10/23/2 | Fri 10/23/26 |
| 79 | Project Complete | 0 days | Fri 10/23/2 | Fri 10/23/26 |

**Figure 18. Project Schedule Summary**

# Results

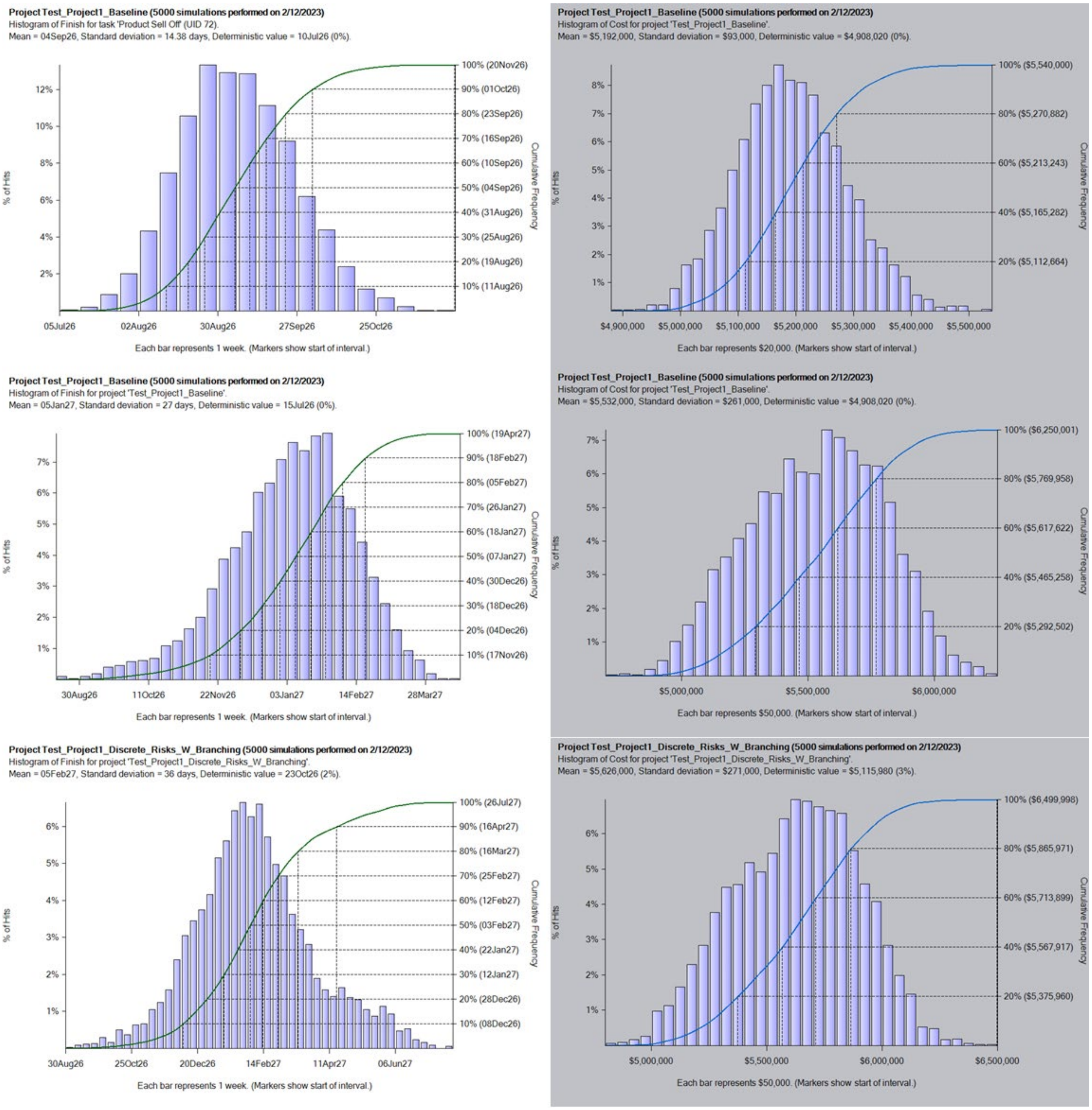


**Figure 19. Project Results for Schedule and Cost**