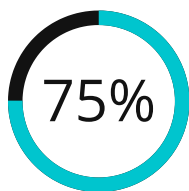
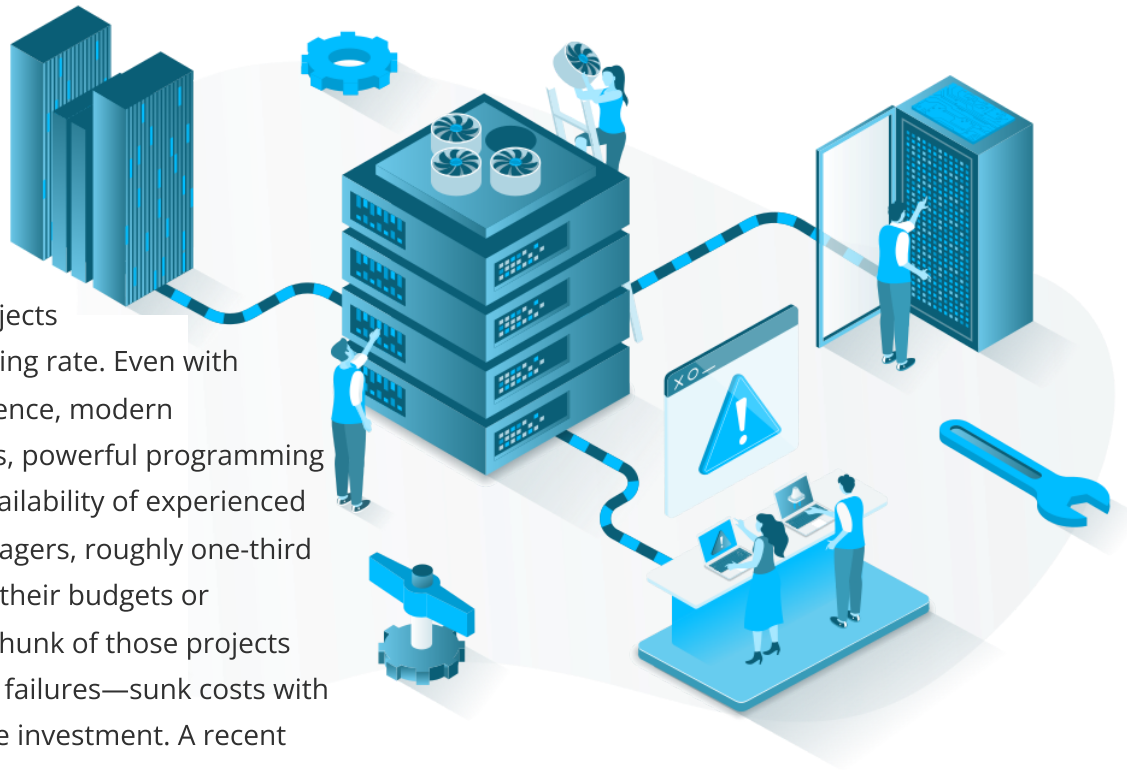


# GUIDE: WHY SOFTWARE PROJECTS FAIL

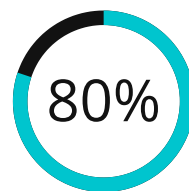


# SOLUTIONS MACHINE GUIDE: Why Software Projects Fail

It's a sad fact of modern life: IT projects in general, and enterprise software projects in particular, fail at an alarming rate. Even with decades of IT project experience, modern development methodologies, powerful programming languages, and the ready availability of experienced developers and project managers, roughly one-third of software projects exceed their budgets or timelines (or both). A good chunk of those projects are abandoned as complete failures—sunk costs with no hope of any return on the investment. A recent survey of IT and business executives by software consultancy Geneca found some sobering statistics:

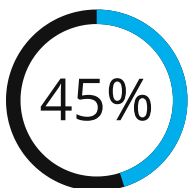


**Of respondents said most of their software projects are “doomed from the start.”**

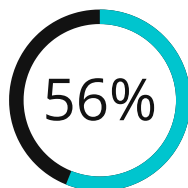


**Said more than half of project time is spent on rework and changes.**

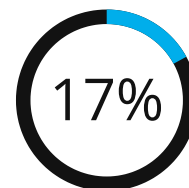
Business consulting group McKinsey also conducted a survey that found:



**Of IT projects go over budget.**



**Of projects fail to deliver the value initially promised.**



**In the worst-case scenario, 17% of IT projects fail so thoroughly that they threaten to put the company out of business.**

# Causes of Software Project Failure

What are the causes of software project failure? There are numerous culprits, but most failures can be traced to a small set of issues.

In the Geneca survey mentioned previously, respondents cited lack of stakeholder involvement, unclear business objectives, and disagreement on the “definition of done” as primary causes of project failure.

A separate McKinsey survey asked a similar set of respondents to identify the top causes for their software project failures.



## The top three were:



**SHIFTING ORGANIZATIONAL PRIORITIES**



**CHANGES IN PROJECT OBJECTIVES AFTER PROJECT START**



**INACCURATE REQUIREMENTS GATHERING**

**OTHER IMPORTANT CAUSES CITED INCLUDE UNCLEAR PROJECT VISION, LACK OF RISK ASSESSMENT, AND INACCURATE COST AND TIME ESTIMATES.**

Notice that none of these have anything to do with the technical skills of the developers, project managers, or other project participants. You usually can't pin a software project failure on bad code. Rather, poor outcomes result from poor understanding of what needs to be done.

# The Wrong Way to Prevent Project Failure

*"IF WE ONLY HAD MORE BUDGET OR TIME, WE COULD MAKE THIS PROJECT A SUCCESS."*

That's a common sentiment, and a common misconception. The truth is that simply throwing more money or time at a project won't make it succeed. Here's why.

Although woeful underfunding of software projects does happen, according to these survey results it is not a common occurrence. To the extent that projects are underfunded, it's a result of the other shortcomings. If you aren't clear on the fundamentals such as business objectives, risks, and opportunities, creating accurate time and cost estimates is almost impossible.

And even in some fantasy world where you could get unlimited money and time for a project, without the fundamentals in place, you would just be throwing good money after bad.

The project would go on forever and never help the business do anything.

Most business executives have seen enough software project failures to know when a project is hopelessly off the rails and will be more inclined to kill the project rather than waste any more money on it.

So what's the answer? Enterprise software, in particular software that must integrate with existing systems, is by definition a complex undertaking. How can you improve the odds of project success?

At Solution Machine, we find that the key to success is conducting a thorough discovery process before embarking on a software project. In project discovery, all those fundamental aspects are defined, documented, and agreed upon by all stakeholders.

## The Discovery Process

The goal of discovery is to assemble important information about the project so that all stakeholders can understand the vision, goals, and scope of the project. All enterprise software development projects should start with a discovery phase. It's especially important for projects that involve:

- **Business-critical applications**
- **Applications that target a highly competitive environment**
- **Applications that require integrations with other systems**
- **Customer-facing applications**
- **New-to-market applications**
- **Applications that are intended to solve a unique business problem (that is, no commercial off-the-shelf solution exists)**

Because of its critical role in project success, discovery is an investment, not a cost. Skimping on the discovery process might provide the illusion of saving time or money, but it will likely come at the price of project failure.

## Project Discovery vs. Product Discovery

**THERE ARE TWO TYPES OF DISCOVERY:  
PROJECT DISCOVERY AND PRODUCT  
DISCOVERY. BOTH ARE ESSENTIAL  
TO PROJECT SUCCESS.**



The goal of project discovery is to identify the high-level business problem (or problems) to be solved and determine the scope of the project. Typical participants include the project sponsor and other decision-makers with knowledge of the business problems, objectives, and budget.

Product discovery is more detailed and relates to the requirements and use cases for specific applications or modules. A software project may involve more than one product, so there might be several product discovery exercises. Product discovery should always be conducted with reference to the findings of project discovery—for example, the scope of a software product should not be outside the scope of the project and should not attempt to solve business problems other than those identified in project discovery.

**PRODUCT DISCOVERY SESSIONS  
SHOULD INVOLVE THE END USERS  
WHO CAN DESCRIBE THEIR PAIN  
POINTS AND WHAT THEY NEED TO  
DO THEIR JOBS MORE EFFICIENTLY.**

For both project and product discovery, sessions should be led by an outsider with experience in the discovery process, who can look at the situation with fresh eyes and ask the right questions.

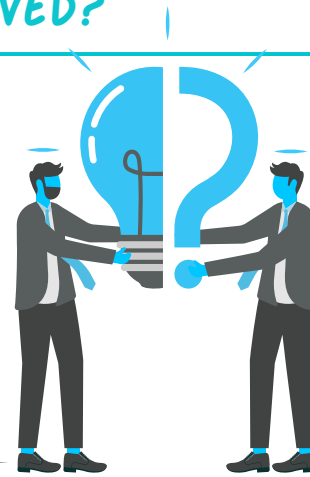


# How Discovery **Is Done**

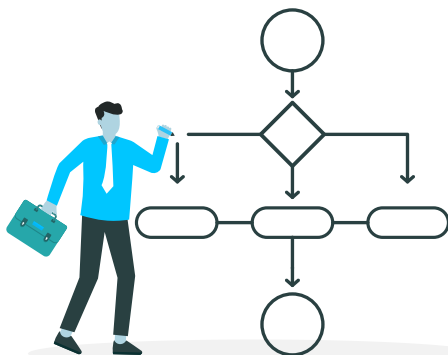
Discovery is conducted as a series of sessions between the project team and client to understand the client's business, goals, strategy, operation process, and other important information. It is the act of gathering essential project information to gain a high-level understanding of the project. In most cases, this is done by getting the answers to specific questions, such as:

## WHAT IS THE BUSINESS PROBLEM (OR PROBLEMS) THAT NEEDS TO BE SOLVED?

These should be stated with reference to one or more business processes and not just complaints about the current software. "It takes too long to complete process X," "Process Y is experiencing too many data entry errors," and "We have too many re-work production orders because of quality issues" are all examples of business problems.



## WHAT IS THE EXPECTED PROJECT OUTCOME?



This is the benefit the business expects to realize by implementing the software solution, expressed best by reference to one or more key performance metrics. It can involve reducing the number of user or customer complaints, reducing production cycle times, reducing employee turnover, and so on.

## WHO IS THE PROJECT SPONSOR?

A sponsor is a person or a group that provides executive-level support and resources for the project. The sponsor can influence the project and should be included in the stakeholders' list.



## WHAT IS THE PROJECT'S TIMEFRAME?

The client may have a scheduled date by which the project should be completed. In some cases this may be dictated by legal or regulatory requirements.



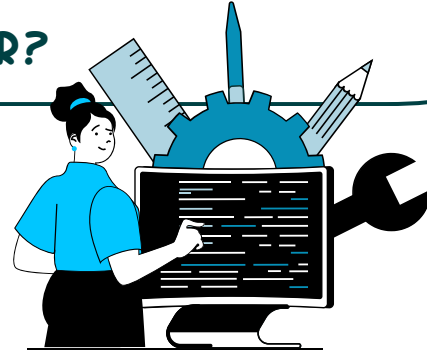
## WHAT IS THE BUDGET FOR THE PROJECT?



This is one of the major constraints that will affect the extent to which expectations can be implemented. After the requirements are identified and prioritized, a cost estimate can be prepared and compared with the budget.

## WHO IS THE END USER?

This may or may not be the same as the project customer. The end users are the people who will use the software to do their jobs, and they must be involved throughout the project.



At the end of the discovery process, the project team and client stakeholders should have a solid understanding and common vision for the project, as well as a list of high-level product requirements or “user stories” that describe what, in the ideal case, the software solution should do.

## Applying the Discovery Results

*FIRST OFF, THE STAKEHOLDERS WILL TAKE THE LIST OF HIGH-LEVEL REQUIREMENTS AND PRIORITIZE THEM, USUALLY WITH THE MOST IMPACTFUL ITEMS AT THE TOP. YOU WILL ALMOST NEVER BE ABLE TO INCLUDE EVERYTHING WITHIN THE CONSTRAINTS OF THE PROJECT BUDGET, SO SOME ITEMS WILL NEED TO BE DEFERRED TO A LATER PROJECT.*

After some discussion and negotiation, the list of items that is included in the project scope should be documented and agreed upon. This provides the team and stakeholders with two important advantages:



THEY HAVE A BRIGHT-LINE DISTINCTION BETWEEN WHAT'S IN SCOPE AND WHAT'S NOT. NO ONE SHOULD BE WORKING ON ANYTHING THAT'S OUT OF SCOPE.



THEY HAVE A COMMON UNDERSTANDING OF WHAT "DONE" LOOKS LIKE.

Too many projects fail because they don't have these principles guiding them (or the principles aren't enforced). Without them, projects go on forever (or until the money runs out) with items continuously added to the project scope and no definition of when the project is complete.

## How Solution Machine Can Help

The risks of skipping the discovery process, or of not doing a proper, thorough job of it, are immense, and echo the causes of project failure mentioned at the beginning of this guide. They include, among others:

- Constant scope creep due to poorly defined success metrics
- Out-of-control costs due to re-work and changes
- Missed deadlines because project boundaries are not appropriately clarified and enforced
- Unmet expectations due to misunderstanding and lack of focus

Because the discovery process is so essential to project success, it should not be left to chance. The team conducting the discovery should be well qualified with wide-ranging experience.

That's where Solution Machine can help. Our consultants have many years of experience leading software projects in a wide variety of industries. Our tried-and-true discovery process has been sharpened by long experience. Conducting discovery and applying its results is second-nature to us; it guides everything we do in a software development project.

To learn more about the importance of software project discovery and how Solution Machine's discovery process can keep your project on track, contact us today.





## TALKING TO US ABOUT YOUR PROJECT IS EASY.



737.757.0967



[www.solutionmachine.com](http://www.solutionmachine.com)



[team@solutionmachine.com](mailto:team@solutionmachine.com)



1320 Arrow Point Drive,  
Suite 501  
Cedar Park, TX 786133

[Solution Machine](#) is a software development company that values its role as your strategic business partner. We are committed to helping our clients achieve their goals through our consultative process, proactive and transparent communications, design, implementation, and ongoing support services of their business technology.

Our team of dedicated professionals is passionate about taking responsibility and accountability for your success. We will do whatever it takes to get a successful outcome for your business.

For more information on how Solution Machine can help with your custom or hybrid digital transformation project, [contact us today](#).