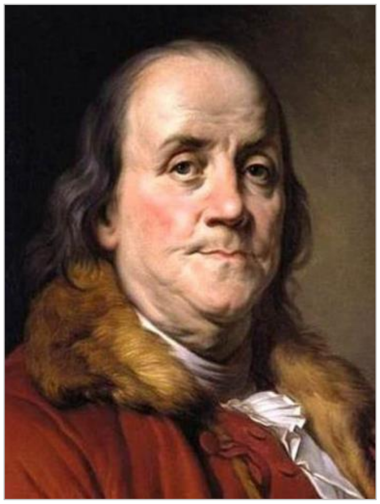
# A Planning Guide for Software Projects

*Six Basic Steps To Insure Success*

An Executive Overview



*"By failing to prepare, you are preparing to fail."*

*Benjamin Franklin*

Jerry Horne

**By Jerry Horne**

# A Planning Guide for Software Projects

## *An Executive Overview*

When asked for guidance by busy executives, often with limited experiences in software project preparation and development, how best to plan for a software implementation, migration, or upgrade project, I am happy to provide practicable and sound advice based upon its twenty plus years of cumulative project experiences.

While every project may have its unique needs and requirements, I have found that when these six commonly shared steps are properly addressed, software projects tend to travel a smoother path. Plus, projects seem to have a quicker ROI, and a better chance of being successful, if the suggested attention-to-detail can be achieved at each step.

- **Step One** – *Build a project team to match project needs*
- **Step Two** – *Formulate a detailed blueprint for your team*
- **Step Three** – *Set reasonable deliverables, timelines, and milestones*
- **Step Four** – *Create a budget and set an ROI target goal*
- **Step Five** – *Establish reporting guidelines and change processes*
- **Step Six** – *Assess risks if project goals and deadlines are missed*

My recommended project approach is one of applying best case traits and proven methodologies, which can be emulated every time to create repeatable and predictable project results. Successful software projects are never open ended; and, they are always framed by engagement delivery tasks that are due within a specified amount of time, and within a specified budget. Please note that the above steps are not all inclusive for every project; and, because projects may vary in goals, scope, and complexity, additional steps may be necessary or required.

# STEP ONE

*Build a project team to match project needs*

## Building a Team

Software project teams vary in membership size and skill sets, all built around the complexity, scope, and timing of a project. Core ERP project teams typically start with 5 to 10 members. However, for many projects, it may be common for teams to dynamically expand and contract as functional and technical phases may be addressed and completed.

Team member roles, and their skill sets, are often designated as being either **functional** or **technical**. Functional roles are associated with specific business processes or activities; whereas, technical roles may be associated with general IT programming and/or technology processes or activities. In some cases, technical knowledge can also apply to knowledge of technical manufacturing processes for specialized packaging systems, including aspects of mechanical engineering, product conveyance, industrial plant process controls, and the like.

Regardless of a project's scope, there are two required roles that every project team needs:

1. An **executive sponsor**, who champions the project, guides it through a budget approval process; and, often has the most to gain with its successful conclusion.

2. A **project manager**, who manages the team's day-to-day work effort, coordinates resources, provides periodic project updates to the executive staff; and, has decision making authority to keep the project on track.

Depending upon time constraints of the project team, other project demands and commitments, or the availability of internal resources, outside consulting teams, are often utilized as partners to provide supplemental or complemental functional or technical expertise that augments or supports the needs of a core project team.

# STEP TWO

## *Formulate a detailed blueprint for your team*

**Blueprints**

Every project needs a detailed blueprint. Whether one is a construction project manager, or one is a software project manager, both jobs require clear, concise, and detailed instructions to insure a successful outcome. And, for every project, one must have well-defined, achievable, and measurable goals to gauge the final outcome of the project.

The more detail that can be captured upfront will lessen the chance of project creep, and budget over runs; plus, the proficiency of a team's upfront work effort can maximize a projects' outcome and its potential ROI results.

Often project stakeholders – executives and/or power users – have unique goals that are specific to their position, their department, or for their area of responsibility. All collective goals should be documented within the project plan, prioritized, and agreed upon by the team prior to the project starting.

**Project Reasons and Characteristics**

The need or reason for a project may be the result of valid internal requirements *(specific business needs)*, or external requirements *(specific legal reasons or cost-of-doing business)*. Whatever the reason, there should be ample supporting details in the planning documents.

Aside from goals, projects may be influenced in some form by budgets, resources, events, activities, and time. Each of these should be clearly detailed in the project's blueprint.

One universal goal for all team members and stakeholders should be to have no surprises, hidden costs, or unexpected results at the end of the project.

# STEP THREE

## *Set project expectations with reasonable deliverables, timelines, and milestones*

**Project Definitions**

Software projects may be commonly categorized as technical, functional, or transformational. Additionally, a project's complexity may be characterized as being low, medium, or high. How one establishes and sets project expectations may be methodically similar for each; but, every project will have its own unique nuances and strategy.

**Technical** – *Characteristics:* Mostly there are retained business processes; but, there is a change to the technical environment. Moving from an on premise IT data center to a hosted Cloud environment could be an example of a technical project.

**Functional** – *Characteristics:* Mostly there is a retained technical environment; but, there are new or updated business functions to address changing business needs. Installing a CRM program to track customer information is an example of a functional project.

**Transformational** – *Characteristics:* There are significant changes to business functionality because of changing or evolving business needs; and, there may also be accompanying modifications to the technical environment, usually to address new business issues. For example, mergers and acquisitions often require transformational software projects to blend new companies together, especially if both have different ERP systems involved.

**Low Complexity** – *Characteristics:* There is a limited customization effort, if any; a low number of modules are involved; impact to business process/changes are limited; low staffing resources are required; duration is less than a few months; it is a low risk project; and, an ROI is projected to be less than 1 year. Installing an ERP technical tools upgrade, or installing a new report writing program are examples of a low complexity project.

**Medium Complexity** – *Characteristics:* Some customization or personalization effort may be required; an increased number of modules are involved; more business process/changes are impacted; increased resources are required; duration may be six to twelve months; it is a medium risk project; and, an ROI may be projected between 1-2 years. Installing a new human capital management (HCM) program would be an example of a medium complexity project that can touch multiple programs within a company's ERP environment.

**High Complexity** – *Characteristics:* There is significant customization or personalization effort required; a large number of modules are involved; significant business process/changes are impacted; many resources are required; duration may be twelve months or much longer; it is a higher risk project; and, an ROI may be projected between 2-5 years. Implementing an advanced warehouse and logistics tracking program to augment an existing distribution software environment, or implementing a new brand new ERP program with many edge program applications would be examples of a high complexity project.

## Setting Expectations with Well-Defined and Reasonable Deliverables in Mind

Never bite off more than one can chew, goes the old adage. Software projects should follow suit by setting expectations outlined by easy-to-understand parameters, well-defined tasks, and measurable project deliverables. For all software projects, their tasks need to be prioritized, staffed appropriately, sequenced properly, and thought-out in advance to make the best use of resources, time, and budget.

To illustrate, in a construction project, a concrete foundation is an essential first step before starting the building's framework. But, before the concrete is poured, site surveys must be conducted, boundaries must be fixed, wooden or metal forms must be erected, and the site must be physically prepared before starting the concrete work. Similarly, for software projects, before any software can be configured, tested, and verified, a site survey and discovery to understand one's core business processes, and any impacted business procedures, is an essential first step in the process to define, design, and plan a software project. And, if there is any integration required with other programs, this integration effort must be detailed and planned in advance.

## Timelines

Timelines can be difficult to estimate and plan for three reasons. (1) As much as one may try, some things can influence projects that are totally out of one's control. Business priorities can change; economic conditions can change; and, operational circumstances can change. (2) For complex projects in some organizations, one may end up sharing staffing resources with other prioritized projects, which can cause delays. And, (3) project constraints can be negatively impacted by the following: a project's length of duration; any shift in budget finances; and, and unforeseen demands on team members. Projects may need to factor in adequate time for holidays, vacations, and normal business activities; as well as, a team member's regular job role and responsibilities.

A project's complexity always impacts a project's duration. Generally speaking, the larger the scope of a project, the longer it will take to be completed. I am often asked if there is a way to compress the amount of time required. The answer is a highly qualified maybe.

By increasing the size of the team, one may be able to shorten some aspects of a project. And, by lessening a team member's regular job responsibilities, one may free up more time that can be spent on project tasks. But, there are usually points of diminishing return that one can reasonably control. Increased resources, and focused time, may only improve certain parallel project processes. Sequential processes may need to be closely monitored.

### Milestones

Milestones chart the progress of a project's work effort along a planned course of action. One must be mindful of common project constraints, which are identified as (1) **staffing availability** *(people),* (2) **resource availability** *(materials, facility access, technical, mechanical)*, (3) **time mandates** *(to meet a specific date, to comply with a business requirement, or to meet a legal compliance target)*, and (4) **budgetary** *(which is frequently re-evaluated quarterly to review and align with revenue attainment).* Changes to any one of these four constraints can both negatively and positively influence the others. Increasing staffing might reduce the time required to complete a step; but, it may also negatively impact a staffing budget goal. Conversely, reducing staffing may positively impact a staff budget goal; but, it may negatively stretch out a timeline causing a missed milestone date.

For complex projects, there may be advantages to using smaller, more manageable milestone steps, with each smaller step having its own unique milestones goals.

# STEP FOUR

## *Create a budget and set an ROI target goal*

**Estimating Cost vs. Actual Cost**

Two common executive questions I hear are: (1) what will be the software project's cost? And, (2) how quickly will there be a return on the investment? These are very specific questions. To provide specific and accurate answers, these questions require an in-depth review. And, there will likely be a considerable time investment to discover and to uncover specific needs, to establish and prioritize goals, and to set project budgets.

However, for some software project discussions, this level of work detail may be like putting the cart before the horse. A potential software project may first need to navigate through an internal vetting process before it can be included in a formal budgeting discussion; and, perhaps, for vetting purposes, only a general idea of project costs may be necessary.

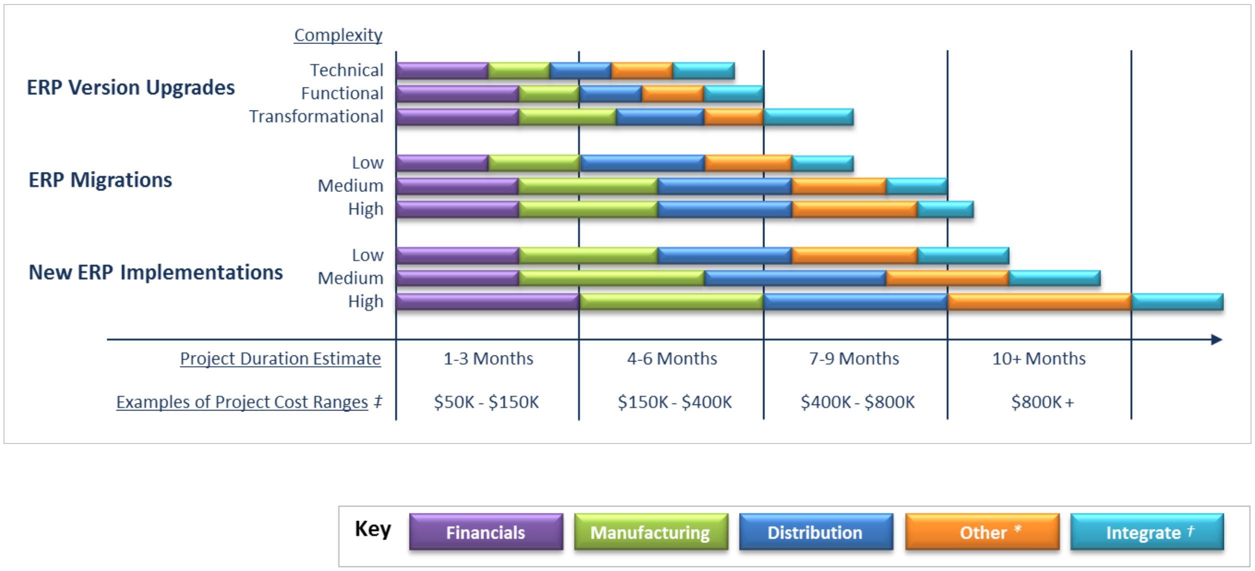
Based upon my experiences, what executives may be really seeking is the answer to this general type of business planning scenario: *What might be the "typical" cost range and time required for a similar sized project to ours, so we can plan for its impact on the rest of our company's budget; help estimate the required resources needed; and, to project IT staffing needs? And, then, based upon these general estimates, maybe we can determine if the project is affordable, or cost justifiable so it can move forward to a formal budget process.* Whew! Does this broad scenario seem familiar? I hear it often.

There is an inherent difficulty with trying to address "what if" planning scenarios like the one above. Questions can be really awkward to answer, and often answers are inaccurate because, as noted earlier, no two projects have the same blueprint or the exact same complexity. In reality, there is no such thing as a "typical" project. Each one is unique.

## Complexity and Project Duration

I can definitively say that a project's complexity will correlate directly to both time requirements and to budget costs. For planning and vetting discussions, one can make a few educated assumptions by using the chart below to guesstimate a project.

To visually illustrate complexity, the graph shows three different ERP software project types: an upgrade, a migration, and a new implementation. Note that each project type offers different examples of complexity: low, medium, and high. At a glance, as one might expect, an ERP version upgrade in an existing environment should take considerably less time and cost less than a completely new and highly complex ERP implementation.



## Chart Usage Assumptions and Directions

If one's project only pertains to financials, then use just the purple portion of a bar as a reference point. If one's project has financials and distribution, then add the purple and dark blue bars together, and so forth to approximate a project's duration. Then, align the project to its estimated low, medium, or high complexity bars. By combining or subtracting the colored software application segment bars on a line, one may be able to build an approximate idea for a software project timeline and a budget cost range estimate.

Actual cost results may vary significantly due to unknown factors. Please realize that true project costs and ROI cannot be fully identified without an in-depth discovery effort.

### Cost Justification

The justification for a software project may be measured with **tangible** metrics with easily identifiable savings generated by a reduction in errors, or in improvements to labor efficiency. However, the justification could also be measured with **intangible** savings like reductions in customer service calls, or fewer warranty repairs. And, there are times when the justification may be measured by **reducing corporate risk and legal exposure** due to previous product recall liabilities. Finally, sometimes justification reflects various degrees of all of the above for one’s own business reasons. All cost justification techniques may be equally valid.

Every company has their own financial preferences and formulas used for cost justifying a project. Companies can utilize an internal rate of return *(IRR)* formula, a project’s net present value *(NPV)* formula, or calculate the payback in terms of time *(it will take ”x” number of business months/days in savings to pay back the investment spent).*

Additionally, one’s accounting department may have their own return-on-investment *(ROI)* measurement standards for projects, such as if the project expenditure is less than $500K, then the ROI goal should be within 24 months; or, if the project’s expenditure is less than $1M, then the ROI goal should be within 36 months. Or, it may be that an IRR goal should be 20%; or, that an NPV goal should be 15%. Every company sets their own financial ROI goals and targets to be attained.

A budgeted software project and its ROI are often viewed as two sides of the same coin. For ERP projects, there is nearly always a positive operational payback, and a financial ROI that will be attained. The variable may be in how quickly the ROI will be reached and surpassed.

# STEP FIVE

## *Establish reporting guidelines and change processes*

**Reporting**

Executives tell me that a project manager's most critical job function may be project communications that need to be managed in a timely manner; and, one cannot over emphasize its importance to a project team or the company.

Much like a juggler with multiple balls in the air, a project manager must keep team members, and project stakeholders, in the loop on where a project stands with tasks that have been completed, where the project is on its timeline, and how funds are being spent.

If there are project problems to be addressed, then the entire team and stakeholders should be aware of all issues. And, as milestones are met, the entire team should celebrate in their team's success.

There is a balance to be found between "quantity" of reports and "quality" of reports when trying to establish a reasonable project reporting standard. Most project managers find that weekly reports are more than sufficient for their team. However, when a project approaches certain targeted milestones, more frequent reporting may be necessary. This may be especially true in complex projects with a larger team, or with projects having multiple milestones being tracked and attained simultaneously, or with companies engaged at multiple physical sites.

There is a growing trend for project managers to use electronic project reporting and tracking tools such as *Microsoft Project™,* or a similar program. Additionally, I find that many companies have made excellent use of dedicated *SharePoint™* sites that contain all project related information. I highly endorse the usage of these techniques to provide real-time information and project status updates on-demand.

## The Nature of Options

The general nature of all ERP software projects is to explore options that will provide the best case business scenario for a company to move forward.

Executive sponsors, project teams, and project managers commonly learn that there may be multiple ways to achieve the same programmatic outcome. Figuring out which way works best for one's company may be the fun part of a project's design and development phases.

Additionally, teams and managers may discover that some business processes work differently in practice than they do in theory. And, just because one way is specified in a project's blueprint, it does not mean that other options cannot or should not be explored, too. Accept in advance that blueprints will likely change and evolve over time as project teams become better at probing, exploring, and understanding the options presented.

## Tracking Changes

Sometimes requested changes to a project are minor and will have little, if any, impact on the total project budget. However, sometimes, requested project changes can be significant and costly; and, in those cases, there could be a significant impact to a project's budget, staffing, and timeline.

A project manager should formally track and document all requested project changes, including the ones not accepted or approved. The project manager is also responsible for managing the decision process to gain the executive sponsor's buy-in that a proposed change is valid, accepted, and formally approved.

Tracking and recording change requests, and their approval/cost/validation processes, are necessary strategic steps for the overall project goal of not having surprises at the end. In my experience, having a well-documented audit trail for every project change request can be worth the effort. And, if one's company has ISO aspirations, or legal regulatory compliance issues, then a documented decision audit trail can be very beneficial for other business reasons.

# STEP SIX

*Identify risks if project goals and deadlines are missed*

**What If Risk Assessments**

The more significant and important a software project might be for a company, the more time should be spent on identifying and assessing its associated risks.

The risk assessment questions to be explored might be: What happens if the decision to begin the project is put-off? What happens if the project start date is delayed? What happens if the project goes over budget? What happens if a go-live target date is missed? What happens if there is project scope creep? What happens if the project is cancelled? What happens if key members of the project team leave or are reassigned? What happens if the project's goals are met early resulting in sooner than expected business changes?

For each of these questions, one may need to consider both the positive and negative risk impacts of (1) any changes to a company's financial plans or fiscal budget; (2) any changes to other internal operational initiatives or programs; (3) any changes to mandates imposed by strategic business partners, or by regulatory agencies; and, (4) any overall changes felt by exceeding, meeting, or ignoring competitive market pressures.

**The Decision Risk of Now vs. Later**

For complex and/or perceived expensive software projects, companies sometimes struggle with making a business decision to move forward, or even when to start a project. The term "paralysis by analysis" is commonly used in these situations. This may be understandable since everyone wants to make a correct and affordable decision for the right business reasons; but, sometimes specific answers, or a project's impact on business, may not be fully understood, appreciated, or known until one gets deep into the discovery, design, and integration planning aspects of a project. So, how does one decide when and what to do next?

As is often stated, the only true business constant is change. And, business change is always inevitable. One may ask then, how long can inevitable change be put off? Can legitimate business risks and costs be better mitigated and managed today, or could they be put off until tomorrow? Is there a balance point in between? The real business decision factor may come down to a discussion on **moving forward now** or **moving forward later**. Companies, with similar project needs and requirements, may follow completely different paths and make completely different decisions, all for equally valid reasons.

One critical financial consideration, worthy of closer examination in decision discussions, concerns the real impact of a project's calculated return on investment (ROI). As previously noted, ROI is often expressed in terms of time; and, it can be calculated as a monthly savings figure. **It is important to factor in decision discussions that for every month that a project decision is put off, delayed, or deferred, the calculated monthly ROI savings figure will forever be lost, never to be attained or re-captured.**

To illustrate the financial ramifications, suppose a calculated monthly ROI saving figure is $15,000. A project decision delay, or a project start pushed out six months into the future would mean that an estimated savings of $90,000 would be lost forever *(6 x $15,000 = $90,000).* For many companies, this potential ROI savings figure could represent a significant financial number that is difficult to ignore; and, the larger the project might be, the larger the potential ROI saving that could be lost.

Accordingly, for many companies, finding the quickest path forward to enable ROI savings can become a primary motivating factor to start a business project sooner rather than later.

---

## About Jerry Horne

*Jerry has over 25 years of sales/marketing and consultative experience guiding Fortune 500 and Global 1000 executives with strategies that strengthen their business models by using transformative supply chain software. He has worked internationally in Oracle, SAP, and Infor software environments from his home base in Kansas City.*

Version 2.3 12/2016