Software Implementation Checklist



Managing a successful software implementation can be something of an uphill climb for even experienced project managers.

Combining the challenges of project management, change management, and business process management with the additional issues that can accompany any new system, implementation planning requires tenacity, skill, and a collaborative, proactive approach—and it doesn't hurt to have a clear, concise implementation checklist guiding the entire process.

Opt out of the headaches and frustrations that come with a haphazard implementation plan.

Craft a winning implementation checklist by including time-tested essentials, and you'll be able to tick all the boxes that lead to a successful roll out and create engaged and enthusiastic end users.

Why You Need a Software Implementation

Checklist

Big or small, projects that introduce new software to your business environment can send ripples through your entire organization.

End users want to know *why* you're changing as well as how to use the new software.

Senior management wants to know how much money they'll save, or value they'll create, by making the switch.

And from the initial business case and proposal to the final go live date and post implementation review, the project team will have to maintain—and refine, and often defend—a clear idea of its goals for the software implementation.

Flying blind isn't just risky in these circumstances; it may be, in point of fact, impossible.

A successfully executed software implementation plan begins with a detailed checklist.

By creating a formal software implementation checklist, you'll be able to:

- Clearly define the overall project plan and provide a framework for establishing scope, timeline, and budget.
- Set, track, and adjust goals as necessary over the course of the project.
- Establish clear deliverables for every responsible party and each stage of the project, from business case through post implementation.
- Provide concrete evidence of progress for key stakeholders.
- Provide clear benchmarks and context for approvals that require stakeholder review and sign-off.

> *When building your software implementation checklist, it helps to think of the overall project management process as something akin to an old-fashioned caper: to get the job done, you'll need a crack team of specialists from varied disciplines, careful planning, skill and talent, and the flexibility to tackle the unexpected (an intense theme tune is optional but highly recommended).*

Crafting a Successful Software Implementation Checklist

Caveats aside, software implementations are, at their heart, just another type of project designed to improve business processes.

But what's simple in concept can quickly become complex in execution, as any experienced project manager knows.

When building your software implementation checklist, it helps to think of the overall project management process as something akin to an old-fashioned caper: to get the job done, you'll need a crack team of specialists from varied disciplines, careful planning, skill and talent, and the flexibility to tackle the unexpected (an intense and memorable theme tune is optional but highly recommended).

Like the businesses that rely on them, no two implementation checklists will be exactly alike. You can get the ball rolling, however, by starting with one that includes the essentials for success.

Consider this sample checklist, broken out by each phase of the project.

1. Business Case

Before the project can begin, it needs to be approved. And to be approved, it needs to provide concrete value for the business, usually in the form of return on investment (ROI).

At this stage, the project manager builds a case justifying the project to senior management. To get things moving, make sure you:

- **Perform Needs Analysis**
    - Identify the problem or challenge this project is meant to address.
    - Establish the features and tools required to meet this need effectively.
    - Quantify the potential impact on value generation/cost savings/total cost of ownership created by implementing the proposed solution.
- **Establish Software Criteria**

- Identify the purpose of the new software required to complete this project.
- Establish whether this solution augments or replaces existing capabilities.
- Calculate the total time and resource commitment accompanying an ideal solution.
- Calculate the total time and resource commitment accompanying an acceptable solution (i.e., one that meets budgetary and performance requirements for target goals but may not include all features offered by the ideal solution).
- Consider compatibility with your existing software environment and compliance issues.
    - What legacy issues must be addressed or overcome?
    - What measures will need to be taken to ensure the new software and your revised software environment remain compliant with legal, industry, and other requirements?
    - How will the new system integrate with and/or enhance existing data management capabilities?
        - Will the new solution require separate data management?
        - What time and resource commitments will be required to maintain distinct data sets, if any, and how will it impact performance and efficiency?
    - **Identify and Secure Internal and External Resources**
        - Identify which implementation roles can be filled internally and which will be filled by third parties.
        - Define the scope of services provided by third-party participants and alternatives that could conceivably be provided using internal resources.
        - Secure the agreement of internal resources (e.g., technical and functional leads, subject matter experts, management) and provide estimates on the time and resource commitments required to participate in the project.
    - **Present a Complete Business Justification to the Appropriate Stakeholders**
        - All end users have a clear understanding of the

changes involved in the project, and the ways in which the new system will impact existing workflows and business processes.
        - Information Technology (IT) is engaged and has provided the necessary technical specifications and other requirements the new software system must provide to meet compliance standards, system compatibility and operational demands.
    - **Secure feedback and Sign-off from All Stakeholders**
        - End users.
        - Information Technology department.
        - Management from whose budget the project will be funded.

2. Software Vendor Review and Selection

Once management signs off on the project, you're ready to choose a software vendor. At this stage, it's time to:

- **Establish a Vendor Review Team**
    - Build a team made up of:
        - Functional experts.
        - Technical experts.
        - Senior management.
        - End users.
        - Independent consultants to provide additional objectivity.
    - **Establish Selection Criteria Based on Software Criteria, Project Goals, and Budget**
        - Identify vendors who provide solutions that match project targets.
        - Evaluate each potential supplier using a ranking system (e.g. 1-10 or A-F).
        - Set criteria by which each potential vendor will be evaluated. Examples include:
            - Total Cost of Ownership (TCO) related to hardware, software, technical and training support, etc.

- Expertise and Capabilities.
    - Demonstrable understanding of your industry and vertical.
    - Inclusion of industry standards and best practices.
    - Customization options, including add-ons specific to your industry or use case.
    - Ability to fulfill ideal configuration versus essential or acceptable configuration.
- Pricing
    - Willingness to negotiate based on project-specific and long-term business considerations.
- Reputation
    - Specific references from companies with a business profile similar to yours.
    - Compliance and performance history.
- Technical and Customer Support
    - Availability of in-house support.
    - Availability of a dedicated account rep or service specialist familiar with your specific needs and system.
    - Availability of training and educational materials (webinars, multimedia content, documentation library, demonstrations, etc.) before, during, and after implementation.
- Customizable Demonstration.
    - Vendor's ability to provide a customized demonstration addressing your unique business needs and workflow requirements.
- **Select a Winning Vendor**
    - Rank and review all candidates.
    - Refine a pool of potential vendors as necessary.
    - Negotiate with and award a contract to

the winning provider.

3. Project Launch

Once your implementation team has selected a vendor, it's time to build your team and a roadmap for the implementation itself.

The team will need to carefully consider a wide range of factors affecting your business and identify the best approach for engaging your organization, setting up a timeline (with benchmarks and deliverables), and performing effective change management throughout the process.

- **Establish and Organize Project Governance**
    - Create a steering committee or governance board to guide the project and ensure it meets its targets. This committee or board works closely with the vendor and the implementation team to adjust and refine the implementation to meet changing business needs and address unexpected strategic or competitive challenges.
    - This committee or board will typically include:
        - The project manager, i.e. the person responsible for the project's completion and success.
        - Senior business unit management associated with the project.
        - The project management office (PMO), which provides general guidance and standardized best practices. They also provide documentation and manage metrics (such as key performance indicators, or KPIs).
        - Functional leads recruited from within your organization. Ideally, these professionals:
            - Are skilled communicators.
            - Have management experience.
            - Have a deep knowledge of software systems in general and project-specific software in particular.
            - Are committed to a successful project and helping their team, coworkers, and organization thrive.
        - Technical leads, which may come from within your organization (if available) or outside it. Evaluate them

using the same criteria as functional leads, with the addition of advanced technical skills.
    - Change management, in the form of a team or an individual. Change management focuses on improving employee adoption and engagement and minimizing roadblocks created by cultural or procedural changes.
    - An executive sponsor, a C-level administrator with direct and vested interest in the successful completion of the project.
- **Announce the Project Launch**
    - Provide as much context and information as necessary to engage your user base and create positive and enthusiastic support.
        - Include details on timeline, scope, and affected processes, departments, and overall organizational goals.
    - Make the launch an "event."
        - Provide snacks and entertainment.
        - Engage the C-Suite to provide top-down support for the project.
        - Be proactive in providing educational resources for those with questions and concerns.

4. Testing, Deployment, and Process Management

With everyone in the loop, the implementation team can get cracking on this "caper" in earnest.

The goal during initial deployment and process management is to refine your approach for optimal performance, efficiency, and user friendliness while hewing as closely as possible to the project's targets for budget, timeline, and scope.

- **Process Optimization**
    - Functional and technical leads identify all business processes affected by the new software.
    - Implementation team identifies areas of potential improvement

and the impact on value offered by optimization.
        - Implementation team identifies other areas of the business that may benefit from the new software, either through addition to the current implementation or through expanded deployment in the future.
- **Initial Software Build and Configuration**
    - Functional leads submit technical requirements to the technical team.
    - The technical team works with the vendor to customize the software to meet the project specs and business needs.
- **Piloting and Testing**
    - The implementation team establishes conference room pilots (CRP), e.g. demonstrations of the software to test its capabilities and measure user experience and user acceptance.
        - Conduct a primary CRP with subject matter experts, facilitated by both technical and functional leaders.
            - Provide documentation and demonstrate how each process is performed using the new software.
            - Record user results in repeating each process.
            - Document all software bugs and usability issues.
        - Adjust the application as necessary to remove usability issues and eliminate errors.
        - Repeat CRPs and adjustments as necessary to optimize application before final deployment and go live.
    - **User Acceptance Testing**
        - Perform end-to-end performance testing on every business process handled by the software from every affected area within the organization.
        - Record all errors and issues.
        - Adjust the application to address issues and errors.
        - Perform an additional user acceptance test to confirm optimization and corrections.
        - Request formal sign-off or buy-in approval from end users based on feedback.
    - **Cutover**
        - Implementation team reviews all tasks necessary to transition from the existing system to the new system.

        - Tasks are assigned to stakeholders based on their specific roles and capabilities.
        - Examples include:
            - Developing and executing a user support plan.
                - Include short- and long-term processes for both self-guided and assisted support.
            - Service desk training and upgrades to meet user support demands.
            - Perform a final month end in previous system (for accounting applications)
            - Document and publish a timeline for data migration, transaction activity in the previous system, etc.
        - Establish a contingency plan to preserve business continuity in the event of unforeseen technology or other issues during the transition process.
    - **Go-Live**
        - Officially activate the new system.
        - Document user experience issues, errors, and feedback.
        - Provide additional in-person and electronic support during the go-live period to ensure optimal user experience and minimize potential downtime.

5. Post Implementation and Review

Go live was a success; the system's up and running, and the implementation process is complete, right?

Well, not quite. The software's rolled out, but the employees using it are still adjusting, and senior management is still keeping a sharp eye out for the value this project promised to produce.

In addition, even the most ambitiously optimized software solution has room for improvement.

Wrapping up the project is the beginning of two new ones: monitoring how well the new software does its intended job, and identifying opportunities to extract more value and savings from its performance.

Close the Project

- **Project Review**
  - Verify all deliverables were accepted and all approvals are recorded.
  - Verify project is recorded as complete.
  - Verify all budgets and financial reporting is complete.
  - Confirm formal project review has been submitted.
  - Collect completed staff reports and evaluations.
  - Release internal and external resources from the project, including staff and third party consultants.
  - Announce project completion.
  - Store project data for future reference.
- **Continuous Improvement and Optimization**
  - Establish protocols to continue to monitor KPIs established by PMO to measure value creation (e.g., savings, lower TCO, process efficiency improvements, etc.) and identify areas in need of improvement.
  - Continue to engage the user base and request feedback that can be used to further improve user experience and preserve user buy-in.

Execute a Software Implementation That Checks All The Boxes

An intelligent plan, supported by a proactive and collaborative implementation team, can take you halfway to the finish line before you begin.

Take the time to craft your own software implementation checklist, and make sure your new software system produces minimal downtime, maximum user acceptance, and demonstrable, lasting value for your business.

What’s your goal today?

1. Use PLANERGY to manage purchasing and accounts payable

We've helped save billions of dollars for our clients through better spend management, process automation in purchasing and finance, and reducing financial risks. To discover how we can help grow your business:

- Read our case studies, client success stories, and testimonials.
- Visit our "Solutions" page to see the areas of your business we can help improve to see if we're a good fit for each other.
- Learn about us, and our long history of helping companies just like yours.

Book a Live Demo

2. Download our guide "Preparing Your AP Department For The Future"

Download a free copy of our guide to future proofing your accounts payable department. You'll also be subscribed to our email newsletter and notified about new articles or if have something interesting to share.

download a free copy of our guide

3. Learn best practices for purchasing, finance, and more

Browse hundreds of articles, containing an amazing number of useful tools, techniques, and best practices. Many readers tell us they would have paid consultants for the advice in these articles.

Related Posts