

TAYLOR OLSON, PMP

DIGITAL PROJECT MANAGEMENT

The Complete Step-by-Step Guide
to a Successful Launch



1

BASIC TRAINING

If you have a job in the advertising or marketing field you probably work at an agency that produces deliverables that fall into one of these four categories:

1. **Printed material** (catalogs, brochures, posters, manuals, magazine ads, newspaper ads, billboards, etc.)
2. **Digital** (websites, apps, etc.)
3. **Broadcast** (videos, radio spots, television ads, etc.)
4. **Events** (workshops, business shows or meetings, press events, parties, festivals, etc.)

Of course, the list of project types could go on and on, but anything you're working on probably falls into one of those categories. And, the categories often overlap each other. For instance, the event you are managing may require a video or some printed support materials. There may also be a website for registration and event information. This book will focus on managing digital projects. We all know that the technical world moves quite rapidly. Just as soon as we figure out how to manage a certain type of digital project, the technology changes and we're left figuring out an entirely new project. But having worked in this field for several years, I can tell you with certainty that the process doesn't really change—just the projects. The stages of development are always the same, although you might apply them to different technologies and outcomes. For instance, the process I'm going to describe was established

before we had to worry about tablet or mobile devices, but I've been able to easily adapt the process to include them.

TYPES OF RESOURCES

If you're just starting out in your career, one of the first things you've probably wondered while checking out the office may be, "Who are all these people, and what do they do?" Believe it or not, most of the project categories listed above use several of the same types of resources. For example, most of them are going to need a writer, an art director, and a project manager. So, before we get into the details of how to manage a digital project, let's get our bearings first. Here's a list of popular resource titles found at a digital agency, and a very brief explanation of what they do.

Brand project manager or account lead: The brand project manager (PM) is usually assigned to a certain brand full time and knows the clients and their business inside and out. When building the business requirements for a project, the brand PM is a good resource for determining what the client's major concerns are, how they'll respond to certain situations or obstacles, and how to best work with that client and their team.

Technical project manager: Tech PMs often get shifted from project to project and are not usually responsible for sustaining or maintaining any one particular brand. During the development process the tech PM is the lead and is often seen as the bridge between the brand team and the tech team (developers, infrastructure, etc.). On my team we call this person the *rollout manager*. They need to understand every facet of the project from both sides of the fence—business and technical.

Business analyst: This role is becoming more and more popular and defined. Not all companies have business analysts on staff because they look to the technical project managers to handle the role of assessing the situation and documenting the requirements. But a more advanced company will invest in a qualified business analyst (BA) to focus on requirements while leaving the solutions and implementation up to the tech or rollout PM. BAs are good at finding a way to explain technical concepts to the clients and brand team in a way they can easily understand.

Asset manager: This is a key role on the production team. The asset manager receives delivery of assets (images, videos, copy, etc.) from the

creative team and makes sure they match the specifications of what was expected. This job may also involve searching for available images that fit criteria provided by the art director. In addition, it usually involves keeping track of the assets—which assets are secured, which ones we're still looking for, what was received but did not pass quality review, what retouching is needed, and so forth.

Content authors: Some projects use a content management system (CMS) in place of building html sites. Content authors are experts in how to use the CMS to build websites. Content management systems are most often used for large corporations that want to build one main brand website and then duplicate and localize that site in multiple countries and languages. This is where the term *rollout manager* first evolved, by the way—we build it once and then roll it out around the world.

Art director: These creative individuals determine what the screens will look like in terms of colors, fonts, and layout. They keep busy designing the look and feel for anything that comes out of an agency, such as websites, event promotions, email campaigns, or online banner ads. Or, they may be lending their creative minds to research or brainstorming activities.

Copy writer: Not only do they come up with the headlines and copy, they usually come up with the communication strategy as well.

Information architect: IAs are part designer and part strategist. Through research and focus groups they become experts in consumer online habits and patterns. They weigh the client's business objectives and the creative team's designs along with their own recommendations for a unique and engaging customer experience. This is where science and art come together!

Web developer: Although there's a large variety of programming languages, it's safe to divide developers into two main categories; front-end or web programming. Front-end developers focus on the code that brings the art director's vision to life on screen, while web programmers focus on framework and functionality.

Data architect: This is a type of developer who designs and manages the flow of data from end-to-end. Simply put, if the project requires either a data source or data output, the architect will figure out how to pull the data in, filter it, merge it, present it, and/or push it out to a third party.

Search and analytics: Two groups of resources are included in this category—those who specialize in *search engine optimization* (getting people to visit your website) and *analytics* (professionals who track

where visitors came from and what they did while they were there—which buttons did they click on, how long did they spend on a page, and much more).

Infrastructure manager: This important resource manages the hardware and operating systems behind the web services. Servers, disc space, security, and other back-end services are all part of their *domain*. They also administer the different environments available to the developers and user groups. For instance, initial development will normally be done in a *lower* environment, while the testing is done in a *preproduction* environment, and finally released into the *production* environment.

Quality assurance: The quality team is also called the testing team. Once a project is deemed ready, this team takes it and tries to break it. Not *really*, but they do test it against the original business requirements and ensure that the team produced what the client is expecting. They need to test the same experience in all sorts of different browsers and devices.

This is a pretty short list of resource titles found around the typical agency—there are many more! This should, however, provide a fairly accurate list of who's in the room for a digital project.

TYPES OF AGENCIES

There are also many different types of digital agencies:

Digital production house: This is an agency that does nothing but production work, which is rare. The people at this agency do not create assets, and normally have no art directors or copy writers on staff. A production house (or team) takes delivery of the assets from the creative agency and employs a team of developers and content authors who code and produce the end deliverable. Many times the production agency has many other capabilities, it's just that the clients only want to pay for and use the production resources.

Digital creative agency: This is the team of strategists, art directors, and copy writers who decide what the campaign will be, what the desired outcome is, and what it looks like. They employ art directors and copy writers to hand over design files to the production team for execution and delivery.

Social media: Facebook, Twitter, Snapchat, Instagram, Vine, etc.—there are so many different options for social media, and this team recommends the best platform on which to deliver a client's campaign.

They strategize on how to best represent the brand on various media outlets in order to achieve the marketing and advertising goals.

Search engine optimization (SEO) agencies: SEO professionals know how to direct Google and Yahoo users to your website or application. Even though PMs aren't personally responsible for SEO, it's one of the most important things we can do to ensure a project's success, thus, we'll be talking a lot about this category. Why create a website if nobody can find it?

Analytics: A wise man once said to me, "If you can't measure it, you can't manage it, and you shouldn't be doing it." This applies to many aspects of advertising and marketing, and exceedingly so for websites. If we're being paid to produce a website or application, it's probably business-related and the clients have bottom-line results in mind—retail results.

Full-service digital agency: A full-service agency can do everything from television advertising to events and printed materials, and for digital this means they have every category previously listed in one place.

A client could hire each of these agencies individually or they could hire a full-service agency that does it all. But for the purposes of this book, let's say we are PMs working at a digital production house, and the project is to build a responsive website using a mobile first methodology.

RESPONSIVE DESIGN

Not too long ago there were plenty of agencies who would brand and market themselves as mobile application specialists, and that was fine because it was a new platform, a new technology, and we were all still learning how to adapt to it. But these days we have to do it all—desktop, tablet, mobile, and whatever comes next. It's just not acceptable to build a digital project that isn't responsive.

A responsive website is designed to provide an optimal viewing experience no matter what the screen size of the end user. Not only does it need to have a fluid layout and navigation which resizes and repositions itself depending on screen size, it has to also provide a user experience that's both unique and consistent at the same time. Unique in that with a mobile-sized screen we just can't provide the exact same view that one would see on a larger desktop screen. That would be senseless, the screen is too small for that to work. But, it also has to be consistent

because customers shop between devices and we don't want to confuse or agitate them with a mobile phone version of the website that's completely different than what they saw using their desktop computer. Test this out for yourself by visiting the same website on your desktop computer, smartphone, and tablet. Unless you have a huge desktop screen the experience will probably match the tablet version, but I hope the mobile version is different (but similar).

MOBILE FIRST

There are a couple of different methods for how to approach responsive design. Historically, because the desktop came first, mobile was sort of an afterthought to the desktop experience. In other words, website designers would create the optimal desktop experience without much consideration for the mobile users, and then afterward try to jimmy the desktop experience down to size. This led to a period of time where it was considered best practice to make two completely different websites for the same client. So, we'd build them one site to cater to desktop users and then a whole new site which would only be shown on mobile devices. Internet magic has something called *mobile detection* which is basically wizards living in the internet who can tell if the viewer is using a small screen and if so, send them to the small screen version of the website. Anyway, as you can imagine, the practice of building two different websites was expensive, time consuming, and irritating because *just* when we thought we were done, we had to start all over again.

Also during this time period, mobile bandwidth globally was very inconsistent. In the United States and other first world nations it was *anything goes*—we could stream videos and load content quite quickly. The content side was actually trying to keep up with everything the technology side was allowing us to do. But in other parts of the world, even as close as South America, bandwidth was just not there yet and they couldn't handle the same experience. Low bandwidth meant slow page-load times and almost completely ruled out videos. Not to mention that smartphones weren't even readily available in those countries, so we actually had to make a third type of experience for any one website. *Three* different sites? Yeah—the desktop version, the high-fidelity U.S. mobile version, and the low- or medium-fidelity mobile version for low-bandwidth countries.

Today, technology has improved. Low-bandwidth countries are catching up with high-bandwidth ones, and clients everywhere are demanding responsive design. And I don't disagree—we should build one site that adapts (or is responsive) to the viewing environment using proportion-based grids, flexible images, and an altered CSS (Cascading Style Sheet). I really can't explain CSS right now but will attempt to later in the book. But here's the question: do we start with the mobile screen in mind and then expand it to fit larger screens (mobile first), or do we create the optimal desktop experience and then size it down? Well, that's debatable, and I'm not even sure there's one correct answer to that question—it depends on the project.

The reason so many people like the concept of *mobile first* is because if we try to design the desktop version first and then scale down, it's quite difficult. It's a lot harder to start with robust material and figure out what to delete than to start with a small screen in mind and then add to it. I like this approach because it forces us to come up with a clean, clear design which is easy to follow and navigate.

Another more obvious way to solve this problem is to consider the original concept of the project. Of course, if we're building an experience that mostly benefits a person who's in the middle of a big city with only a phone in their hand, we go mobile first and then make sure that the desktop site looks good as well. A good example for this might be a mobile app for ordering car service. People also need to order car service from their homes, but it seems like one shouldn't have to turn on a computer for something like that.

On the contrary there are lots of other websites more fitting for a larger screen, such as financial programs or large purchase items, just to name a couple. Experts don't agree, however. Some say that *all* websites should be developed as responsive, and others say that there are some user experiences that will never work well on a small screen. As PMs, however, this is not our battle. We have other experts in the room who make these types of decisions—people whose job it is to make these calls. Our job is to create and manage the process and keep everyone on task. For the purposes of this book, we're going with mobile first. We'll design first for the mobile users and then expand for desktop.

The process I'm about to describe works for any type of digital project, whether it be a tablet application purchased through the iPad store or a global web experience on all types of devices in all types of languages. How could that be? It's because I'm not explaining how to

develop the code or write the analytics—I'm just providing a list of line items we need to hit for each project. Every project requires code development, and the code will change based on the project. That's an *NMP* in my world—not my problem. (Don't say that out loud.)

I think the best thing to do at this point is to just get on with it so you can see for yourself. Let's make a website! Ready?

GETTING THE ASSIGNMENT

(Office phone rings)

[Project manager] “Hello?”

[Account manager] “Hey bud, how ya doin'?”

[PM] “Great. It's a little cold in here today, but great otherwise. How about you?”

[Account manager] “I'm doing great, too. Say, we have new a client interested in hiring us to build a website for selling spaceships, and we need to get back to them with an estimate on time and costs. Do you have something *in the can* that you can send me?”

[PM] “Well, I'd have to know a lot more about the project before I can provide any estimates. These types of projects can vary widely, as you can imagine. We actually have a rollout process that we follow that starts with an initiation phase where we nail down the business requirements and then provide the estimate.”

[Account manager] “Right, and we definitely want to follow the process and do this thing the right way, but for now we just need ballparks. They just want to know what they're getting into.”

[PM] “Hmmm. Well, can you think of a website we've built in the past that this would compare to?”

[Account manager] “Not really, and that's part of the problem. The client doesn't even know what they want. We need to guide them. So, we thought we'd send them a ballpark estimate along with a generic schedule so we could start to educate them on the process and what's all involved with this.”

I actually get this call all the time. Do you? If you've been doing this for any length of time, I bet you've fielded a few phone calls

like this. The problem here is clear, right? How can we provide a budget for a project we know almost nothing about? And why are we the only ones who understand that this is a problem? Well, the sales team doesn't always seem to get it. But, we probably don't fully understand their worlds either, so let's be nice.

First, remember, you're a professional PM and also a process consultant. When somebody asks you this type of question it's clear that they don't really know what they're asking for any more than their clients do. In my opinion, they may not be verbally asking for my help, but they sure do need it. So my response would be something like this:

[PM] "This project sounds really exciting and interesting—I love spaceships! I think we could send them high-level estimates for building a generic small, medium, or large website, but also explain to them how vastly different the costs can be depending on the business and functional requirements. Make sure they understand that these are just ballpark figures..."

[Account manager] "Oh, of course."

[PM] "... and that we would love to meet with them to review the rollout process we follow. I have a process map I can walk them through, and I can put together some questions that would help me better understand what they're asking for—we should get the IAs on the call as well. Actually, I'd be happy to meet with the internal team first to review the process before showing it to the client so that we're all on the same page. How does that sound?"

With a quick sentence or two I was able to introduce leadership, instill confidence, and initiate a process.

Introduce leadership: Instead of just sending over what the account manager asked for while my brain slowly melted and peeled away from frustration, I decided to change the paradigm. Sending random costs and timelines over would result in our own team selling something we could never deliver on. Once we send over costs and timelines to people, that *is* what they're going to sell and then try to hold us to. Believe me, that will come back to bite us. Instead, I flipped it around and introduced leadership. I met them halfway by sending along some size-based budgets which could cover any range of project scopes, and offered to lead the team by teaching them the process—my process. You can cover

the budget estimates by sending over three somewhat random, but also logical, cost breakdowns—something like this:

Small = \$50k–\$125K
Medium = \$125K–\$500K
Large = \$500K+

It could be anything—work with the sales people on this—it’s really more their job than a PM’s at this point.

Instill confidence: Through offering to lead the team and teach the process, I instilled confidence by letting the account manager know that they’ve got somebody on the team (me) who can speak intelligently to the client. Why wouldn’t they want to show that they have a defined digital process and a master curator of such? To be truthful, if the clients wanted to figure this stuff out for themselves, they would have—they want to pay us to know how to do it.

Initiate a process: Inserting the process at the very beginning gives us control before things turn to disaster. Here’s a circumstance I’ve come across time and time again: the sales team rushes out on their own without subject matter experts to help consult, and promise everything under the moon to the clients. Then they hand that huge ball of mess over to the PM to deliver. “Wait...we can’t do this...especially not for that price....” Ever been there?

In another scenario, I’ve seen sales teams sell multimillion-dollar programs without any detail whatsoever about what the deliverables will be. This situation isn’t as bad, because you still have half a chance of putting together a reasonable scope. But I’m always amazed when clients sign a proposal for unidentified deliverables. I’m both amazed at the sales team for being able to pull that off, and also amazed at the purchasing departments who allow it. Last time this happened to me, the CEO of a company I was working for gave me this huge book of a proposal that was sold to a particular client and told me to “git r done.” After reading through the proposal I was really confused and told the CEO that it didn’t seem to contain any actual deliverables. He just smiled, gave me a wink, and walked away.

THE ROLLOUT PROCESS

Now that we have our project identified (making a website of some sort), let’s call the make-believe company *Jetzen Spaceships*. We really

don't have any information at this point, so what should we do first? The process I'm about to explain has been tested, refined, and tested again. I've been developing this process for years and can confidently proclaim that as a practitioner I refer to it for every single digital project to which I am assigned as PM or consultant. It won't let you down.

Although the process aligns with the process groups found in the PMI® *Project Management Body of Knowledge (PMBOK® Guide)*, all of these steps are grouped into two main categories—*Plan and Define* and *Construct to Close*. I felt it was important to break up Plan and Define into two areas (Initiating and Planning) because there is a clear tollgate after initiation where the client reviews and approves the project scope and budget. Other than that, the appropriate digital development steps are listed either under Plan and Define or Construct to Close. Table 1.1 shows the 30 rollout steps.

I'm a big believer in planning and have witnessed a huge difference in the success of projects where there was a lot of time spent up-front planning things out rather than just going with the flow. The bulk of the work is always in the planning column.

Although the steps are numbered, they're not always done in succession, and that's important to note because when showing all 30 steps to the team, it's a good idea to show the graphic as well (Table 1.2) or risk having the account team and clients freak out at the sheer number of steps. It just *looks* like a long process—but, as you can see from the Gantt chart, these steps often overlap, and when clients see this graphic they're not only assured that the team knows what they're doing, but also that they're doing everything possible to move quickly.

Notice that not all of the 30 steps are included in the Gantt chart—I don't call out things like communication planning or analytics and SEO analysis. These things are not part of the critical path and just need to get done in tandem with the rest of the steps. There are a lot of steps because PMs need to be detailed so we don't miss anything—but really, it boils down to about 11 key milestones. Once we show a chart like this to the team, they tend to calm down and get onboard. You might not even want to show them the entire list of 30 steps—that's up to you.

In addition, development can be in either Plan and Define or Construct to Close. In Plan and Define, we definitely figure out what each of our development tasks are and get them priced out and scheduled, but we don't necessarily start the actual programming unless the clients have already agreed to pay for it. We'll go into more detail about this

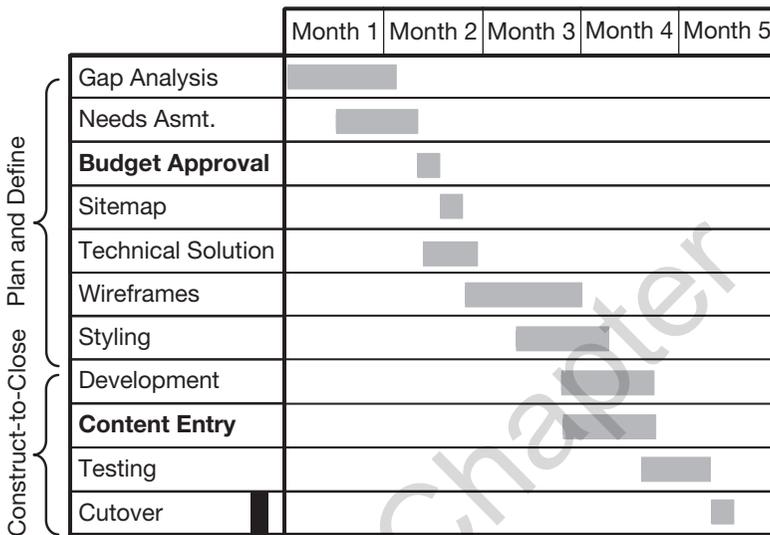
Table 1.1 Rollout process

PLAN AND DEFINE	
<p>INITIATING</p> <ol style="list-style-type: none"> 1. Gap Analysis 2. Workshop 3. Stakeholder List 4. Business Requirements Document 5. Preliminary Budget Estimate 6. Statement of Work (Tollgate) 	<p>PLANNING</p> <ol style="list-style-type: none"> 7. Plan and Define Schedule 8. Communication Planning 9. Sitemap 10. Technical Solution Strategy 11. Wireframes and Functional Specifications 12. Styleguide 13. Analytics Analysis 14. SEO Analysis 15. Infrastructure Assessment 16. IT BOM 17. Development and Change Management 18. Test Strategy and Cases 19. Package Identification 20. Construct to Close Schedule
CONSTRUCT TO CLOSE	
<ol style="list-style-type: none"> 21. Create Content Tracker 22. Asset Quality Review 23. Content Entry 24. Quality Assurance 25. System Integration Testing (SIT) 26. User Acceptance Testing (UAT) 27. Nonfunctional Testing (Performance, Security, Disaster Recovery, Failover) 28. 301 Redirects 29. Cutover Management 30. Transition to Operations 	

later, but for now, take a look at the items that are highlighted in bold in Table 1.2.

Budget approval—This is important for two reasons: first, the clients need to know when they can expect to receive a budget, and second, we need to point out that work will not proceed past this point until the budget is approved.

Table 1.2 Use this Gantt chart to explain how the tasks often overlap each other.



Content entry—This is important because it also affects the creative team. They need to know at what point assets are expected to be delivered so they can allocate resources and get their own budgets approved.

Another thing the clients like to hear at this point is that the team will have a *transition-to-operations* period immediately following cutover. This is generally a two-week period where the build team stays on the project and handles any lingering issues or bugs, and are also there for support, just in case something unexpected happens. Once the site is considered stable, I always prepare some training for the sustain team. (More on this near the end of the book.)

Of course, these steps may change based on the nature of the project, but they're an excellent baseline from which to begin the assessment. Remember also that these steps can be applied to any digital project, whether it's a website, application, or mobile tool. As long as it's a consumer-facing project that requires some sort of cutover activity.

Starting with the Initiation Phase we're going to learn step-by-step how to approach a digital project using Jetzen Spaceships as our example project. It would probably help us to understand a little more about Jetzen before we dive in. Since they're make-believe, it may be difficult to conceptualize what we're doing and why, so here's a make-believe profile on them:

Jetzen Spaceships is a global manufacturer and seller of several different models of spaceships for both public and corporate use.

Headquarters:	Marana, Arizona
U.S. Zone Offices:	Richmond, VA Newton, KS Novato, CA Syracuse, NY
Global Offices:	Mannheim, Germany Shanghai, China
No. of Employees:	19,273
Sector:	Manufacturing
Sales:	\$56.4 Billion 2014

Finally, I'd like to show you how I've organized this book in terms of chapters compared to the process (see Table 1.3). Take a look before we jump right into the initiation phase for our new client, Jetzen.

Let's make a website!

Table 1.3 Contents of book

Chapter	Process Steps Covered
Introduction	
Basic Training	
Needs Assessment	<ol style="list-style-type: none"> 1. Gap Analysis 2. Workshop 3. Stakeholder List
Documentation	<ol style="list-style-type: none"> 4. Business Requirements Document 5. Preliminary Budget Estimate 6. Statement of Work
Communication	<ol style="list-style-type: none"> 7. Plan and Define Schedule 8. Communication Planning
Information Architecture	<ol style="list-style-type: none"> 9. Sitemap 10. Technical Solution Strategy 11. Wireframes and Functional Specifications 12. Styleguide
Analysis	<ol style="list-style-type: none"> 13. Analytics Analysis 14. SEO Analysis 15. Infrastructure Assessment 16. IT BOM
Development	<ol style="list-style-type: none"> 17. Development and Change Management
Preparation	<ol style="list-style-type: none"> 18. Test Strategy and Cases 19. Package Identification 20. Construct to Close Schedule
Content Input	<ol style="list-style-type: none"> 21. Create Content Tracker 22. Asset Quality Review 23. Content Entry 24. Quality Analysis
Testing	<ol style="list-style-type: none"> 25. System Integration Testing (SIT) 26. User Acceptance Testing (UAT) 27. Nonfunctional Testing (Performance, Security, Disaster Recovery, Failover)
Cutover	<ol style="list-style-type: none"> 28. 301 Redirects 29. Cutover Management 30. Transition to Operations