

Planning For A Successful Web Application Development Experience

It is quite challenging to estimate expenses and timing for an application development project before appropriate planning is completed. The challenge for developers is that planning consumes quite a lot of time—up to 20% of the total development hours for most sophisticated applications—and without it, accurate cost and time estimates can't be provided. The challenge for the prospective client is pressure to provide budget estimates before the project is sometimes even scheduled. At Simplicity, we don't have a magic bullet for how to get planning done without good, old-fashioned elbow grease, but we do have a great process. Our process consists of negotiating requirements (what an application will do for its audience) and documenting functional specifications (how the application will work for its audience).

Some clients deal with planning by issuing a Request for Proposal (RFP), but it is the rare RFP that does a comprehensive job with the details needed for accurate estimates. In addition, many RFPs tend to list potential solutions rather than problems or opportunities. It is good to realize that even the best business visionaries often don't know exactly what they need until they start talking about it with developers; perceived solutions are always limited by relevant experiences.

Whether or not you issue an RFP, consider whether you are willing to start a project that will take significant time and money without a clear understanding—by everyone involved—of what is to be accomplished. **Planning not only articulates what will be accomplished, it provides the best opportunity to ensure consensus among stakeholders in what the criteria for project success are, and exactly how the vision of the project will be fulfilled.** There is a very high failure rate for software projects, and many of these failures can be traced back to inadequate planning. Inadequate planning makes it almost impossible to measure the quality of software that is developed, because measurement benchmarks won't exist.

Getting Started

Application development is no different than the development of any complex system. Those who have been through the reconstruction of a kitchen or bathroom may have horror stories to tell about the process. And the vast majority of those stories could have been avoided by working with experienced architects or planners. Experienced planners will start by asking questions about problems or opportunities you have identified. They will try hard to stay away from discussions about solutions until these problems and opportunities are documented.

The Risks of Planning

Analysis paralysis—too much planning—can be as harmful as no planning at all. You have to be willing to accept the risk, once your planning effort is over and the coding starts, that there will be changes. The best planners divine 80-90% of a project. Can you accept 10-20% of a project as a cost or time overrun? It is a whole lot better than the alternative.

What Are the Contents of Requirements and Specifications Documents?

These are the planning categories Simplicity develops with our clients. Within each category, there are requirements (or 'business rules' or 'features') and functional specifications. Sometimes sections are very long, sometimes they are not, but in every case, we address each category as thoroughly as time allows. It can take as little as a day and as long as a month to research these categories.

Audiences: who are the audiences for the application? The more detailed your description or persona, the more successful your experience will be. Careful audience analysis provides the background needed by the development team to make smart decisions. Decisions should always be made by considering what is best for the audiences that will use and support the application.

Use cases/scenarios: how will the audience use the application? You can't just think about these, you have to write them down. Writing them down drives consensus and creates brainstorming opportunities.

Business processes: how will the application change the way business is done? Many software projects fail because the users are not prepared for the behavior changes the software will drive.

Definitions: a list of terms and phrases used in the project. Avoid the conversations that start: "I didn't realize that was what you meant by that!"

Environments: testing, staging, production, systems software and hardware, networks. Don't be fooled into thinking the only cost of your project is in software development.

Versioning: product and software versioning policies. This is always important, but especially if you are developing product that may be in production in multiple, supportable versions.

Transaction flow: how will the results of any transaction flow from screen to screen and component to component? This step may also include preliminary sketches of **user interface** screens.

Reporting and auditing: what will be tracked and how will it be reported? Tracking what happens with your application will help you plan for new features, bill correctly, and solve problems faster.

Capacity and performance: how will it be monitored, and against what criteria? Your application can look great, and meet all sorts of business imperatives, but if it isn't reliable you'll be in support purgatory. Make sure you define what 'reliable' means.

Support: how will problems and changes be handled both during and after the project? Will you have bug and change tracking systems? They are usually software-based, and may take some training to use efficiently.

Launch process: who needs to be a part of it, how will communications be handled, how will training be provided, and when will it go live?

Access Rules/Security: authentication, authorization, encryption, privacy, non-repudiation.

Here is an example of a requirement for this section:

Users must authenticate before gaining access to the application. They will be automatically logged off after 15 minutes of inactivity.

And a functional specification:

Users are identified by their email address, for the purpose of logging in to the site. When a new user is created, a randomly assigned password is emailed to the user. This password provides access to the change password screen only. Once the user has changed their password they have full access to the application. The user-supplied password replaces the randomly assigned password. "Inactivity" is measured from the last keystroke.

Technology: It is likely that specifications will include a description of the technology being used to implement the project, since the technologies used will have an impact on the final cost of the project.

Once the coding starts, it is important to have a process in place that helps manage written changes to the specifications. Once the project is complete, this document will be used to help bring new project team members—or potential investors—up to speed on what has been accomplished.

For more information, or to schedule an appointment, contact Eric Hughes,
Complexity's Managing Director: 510.763.4488 or eric@complexity.net.