White paper

Plan your development plan

An outline of a product development plan

www.stdtime.com
Scoutwest, Inc.

Plan your development plan

When your company approaches clients to do large projects, you may be asked questions like: “What development tools do you use?” “Do you use a methodology to guarantee quality?” “How many people do you have on your team?” “How are they broken down by technical discipline, and training?” These are some of the same questions newly hired staff, investors, partners, and company executives will want to know the answers to. The answers are contained in your development plan.

Like a business plan, a development plan includes full disclosure of your development organization. The plan usually only needs to be written once, and then updated when changes occur in the organization. Derivative plans may also need to be written for specific clients or projects. These derivative plans would likely be more specific in detail. As with a business plan, a development plan can be effective in communicating your intentions to other people. A succinct document that contains all your development details can convey areas where you feel strongly, and areas where some leeway is allowed. It lays out a vision, and helps people of like minds to work together. Developers, QA, technical writers, managers, executives, and other stakeholders are able to get on the same page with your vision. It even helps you remember why you feel strongly about certain things. It’s like a mini business plan for your development department, but without the sales, marketing, and financial aspects. You may consider communicating certain aspects of the plan verbally during company or divisional meetings. This can help circulate the vision you have for the group, and if it’s done in a non-preachy way, people will accept it and get on board with it. It might even help to get them to read the document in the first place. Here are some possible things to include:

1) Development methodologies

Include the details of any formal or informal development methodologies you use. Communicate areas where strict adherence is expected, and areas where personal choice is okay. List policies and procedures that are helpful to improved efficiency.

2) Languages and technologies

Describe technical languages, and technologies you will use for development. Include the advantages and disadvantages of these choices.

3) Development tools

List all the approved tools that may be used, and the current versions you are using. This ensures that all team members are using the same tools. Indicate where each tool is located, and the number of user licenses you have purchased for your departmental use.

4) Project tracking

Describe tools and procedures used for tracking time and project status for engineers, managers, and executives. Describe error factors, earned value, and other aspects of tracking project status.

5) Issue management

Describe policies and roles for issue flow of defects, enhancements, and other issues that come up during development. Use this section to ensure that critical issues are not forgotten or ignored. Describe how this affects version and release management.

6) Product delivery methods

Describe your philosophy on release quality, prototypes, alphas, betas, release candidates, and gold releases. Describe how releases are made available to testers and customers. Include your manufacturing steps and partners used for each part of the product. This may also be an appropriate place to describe build processes used to assemble finished releases of the product.

7) Interaction between departments

As development proceeds, there are natural interaction points between development, QA, and documentation people. Describe when and how these take place for optimal project efficiency. Describe strategic departmental meetings that keep the process on track.

8) Coding conventions

Formally document your expectations for the style of writing software code. The biggest benefit of consistent coding conventions is that is makes the product appear to have been written by a single author. Coding conventions also remove differences that make reading code more difficult. Simplicity and predictability in coding style is important. After all, these are your company's intellectual property assets you are protecting.

9) Source control

Name the tools and methods you will use for source code version control, that is, the way you will store iterative works of your product source code, and retrieve them upon demand.

10) Backup and restore procedures

Describe your procedures for backing up critical intellectual property, and your means of restoring from backups if necessary. You may even want to describe an offsite backup plan in the event of major catastrophe.

11) File formats and file versions

File versions and formats can sometimes be a point of discussion when working with clients on a contract basis. Make sure you meet their needs for files you supply them. Enumerate all the file formats you feel comfortable supporting.

About Us

Scoutwest, Inc. develops and publishes project management and time tracking products for consulting, manufacturing, government, and general business applications.

Thousands of small to large businesses, in dozens of countries worldwide, trust their mission critical business processes to Scoutwest products. Standard Time® and Standard Issue® work together to offer well-rounded project management solutions.

We specialize in packaged software for timesheets, project management, time tracking, defect tracking, and issue tracking. Standard Time is a web-based timesheet that also runs on Windows, Palm OS, and Pocket PC. It can be used for client billing and task management. Standard Issue is used for bug tracking and general issue tracking.

Please visit these web sites for more information.

www.stdtime.com
www.sdtissue.com