Strategies and Methods for Gathering Your Requirements During System Development

Gathering requirements for a system is critical to obtaining all necessary functionality to meet your needs. This step leads to more complete user acceptance of your new laboratory informatics system. Many techniques can be used to collect appropriate user requirements. Interviews, user stories, and shadowing are three distinct methods to gather requirements, but other hybrid approaches have been used as well. These techniques can be used in either a Waterfall or an Agile approach to software development. Below are strategies and techniques for gathering the proper requirements at the right level of detail to set your organization up for a successful project.

Gathering Requirements

At the onset of a project, one of the most important things to remember is to never assume you understand what a business wants. It is important to begin the investigation impartially with the user base to develop knowledge of the business and of how things are accomplished in the organization. This process helps clarify the scope of the project, which must be clearly documented for all to review and approve. The software solutions should not be the entire end game, because some requirements will not fit into a software solution without modification. Demos can help users refine their requirements. All requirements must be SMART (specific, measurable, attainable, relevant, and timely).

Conducting Interviews

In collecting user requirements, it is critical to capture the workflows and needs of the users, approvers, management, and administrators of the new system. Doing this through an interview process allows each role to explain how they perfom their tasks, and allow for follow-up questions or clarification. From the interviews, workflows of existing processes may be generated to capture the direction and details for how work progresses in the laboratory and among departments. This work can be divided into logical categories or work units.

Once captured, workflow efficiency can be optimized based on the type of system being implemented. Where possible, adjustment of workflows should capture the strengths of the new system and not be confined to automating legacy business processes. The workflows can then be used to establish the actual requirements of the system in a list, which will aid with the implementation of your new system and any testing that may be required in the future.

User Stories

It is possible to generate value by having users explain how they follow their current business process or execute functions in their laboratories. These 'stories" can be translated into workflows and requirements. Sometimes these stories are detailed enough to establish the positive and negative aspects of the current workflow. These stories are also used to better understand interactions of colleagues and departments in the organization. Any challenges uncovered with the information that was gathered can be addressed with the new informatics deployment project.

Shadowing

The user base may not be able to sufficiently explain how they perform tasks or do their jobs, therefore, shadowing may be necessary. An expert on laboratory informatics can follow multiple users around to take notes and screen captures. The screen captures can be used to generate requirements, establish workflows, provide a visual for actual fields and prompts within the system, and help determine priorities within the system. Shadowing also offers the opportunity to ask questions, which aid in the development of workflows and requirements.

Strategies and Methods for Gathering Your Requirements During System Development

Development Techniques

There are two widely used techniques for developing a software solution. One is Waterfall, where it follows a logical sequential flow, and the other is Agile, which follows an iterative process. Each technique is explained better below.

Waterfall

A waterfall model breaks down the deployment of a new informatics system into linear, sequential tasks, where each task depends on the deliverables of the previous one as a stage gate. This is a less iterative and flexible process when creating requirements. It moves in one direction through conception, initiation, analysis, design, construction, testing, deployment, and maintenance.

This approach is very structured, with the stage gates requiring each phase of data collection to be completed before the next phase commences. It starts with generating the plan of activities, followed by the requirements. This can be challenging as it means addressing user requirements for a system that the users have no experience with while moving from one stage of a project to the next.

The waterfall model makes it easy to control the effort and time in each phase. A project manager can regulate the amount of time needed to accomplish an activity before reaching a stage gate. Once a stage gate is met, the project manager can move to the next part of the linear workflow with ease.

WATERFALL METHODOLOGY

CONCEPTION

INITIATION

ANALYSIS

DESIGN

CONCEPTION

INITIATION

ANALYSIS

DESIGN

In a waterfall environment, all three techniques (interviews, stories, and shadowing) can be used to collect requirements once the plan is established. Still, user requirements gathering can become problematic. As an example, frequently when gathering requirements via stories, users may inadvertently change the scope or complicate the project with indecision. Because Waterfall is a stage-gate process, requirements gathering must cease at a defined time and date. A strong project manager can enforce this so that the next set of tasks in the project can commence. Requirements that are captured after this point can be executed as part of a future upgrade.
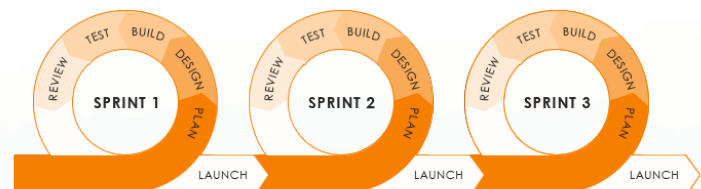
Agile

Agile software development is based on these principles:

- Customer satisfaction
- Frequently changing requirements
- Frequently delivering working software to the users for evaluation
- Daily face-to-face cooperation between the business and developers
- Motivated trusted individuals who are empowered to make decisions
- Success measured by user acceptance
- Maintaing a constant pace
- Continuous attention to good design
- Simplicity to minimize work
- Best architectures, requirements, and designs that emerge from self-organizing teams
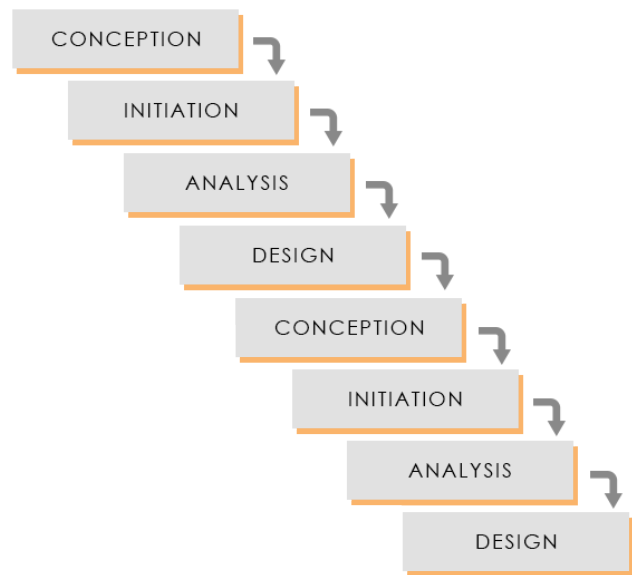
AGILE METHODOLOGY

SPRINT 1   SPRINT 2   SPRINT 3
REVIEW TEST BUILD DESIGN PLAN
LAUNCH   LAUNCH   LAUNCH

Agile is an iterative technique in which some requirements are refined or obtained through interviews or user stories. The system can be partially developed and shown to the users right after the information has been collected. The users have time to comment on the development and give input, so that the software can be refined based on the comments, and then shown again to the user base. This process can be challenging to manage because it is iterative, but has benefits in the speed in which a new system is developed. The process must adhere to a project schedule, or the iterations (sprints) can become numerous and complex. Complete requirements gathering occurs over time, with intermediate demonstrations until the functionality is completed in the system.

Important Considerations

In the requirements gathering process, sometimes additional system changes are needed based upon the requirements gathered. Even though a system can accommodate most requirements, there will always be outliers. These outliers must be prioritized. If a change is determined to be required it must be evaluated as a configuration change, customization, or a combination of both. Configuration changes are easy to execute and maintain because they are typically built into the software. Customizations touch program code and can be difficult to execute and maintain. **You should not base a solution on cutting-edge software only to discover that it cannot easily meet requirements in your organization.** Just because it is cutting edge does not mean it is the right selection for your business.

There are many ways to collect requirements for a new laboratory informatics system. Because there is no one right way to collect them, getting there can be challenging. The methods listed here can be combined into a hybrid technique, which if managed effectively, can lead to a complete gathering of requirements.