
Runaway Product Requirements

Taming the chaos of product management

Less than a generation ago, interpreting market needs to a company's engineers for their guidance in developing new product releases – including complex products like enterprise software – was something product managers could handle using little more than text memos, lists, sticky notes, and spreadsheets. It was demanding work, of course, but it remained well within the grasp of the talent and tools of today.

That was then

Today, companies in industry after industry have fallen under extraordinary stress. And the burden of managing their products' requirements has swollen. Customers have grown more demanding. Market segments have repeatedly splintered and subdivided. Competition has become more aggressive. Cost pressures have intensified. Customer loyalty has declined. Demands for product features come from everywhere. And the pace of product introduction has accelerated dramatically while their life-cycles, in many cases, have been truncated from years to just months. Beyond that, for many markets, the economy remains stuck in low gear.

The need to clearly define the elements of innovation, and to make sure the company's products include the right features, is more important today than ever - both for new products and for new releases of existing ones

Product managers and executives, as a result, have seen their workloads rise exponentially. Their core assignment – setting out clear specifications for future revisions and product innovations – has grown chaotic and barely manageable. Data has ballooned in volume and scattered into a variety of disconnected silos. The links between sources and their requirements have broken. Connections to the company's core business strategies have grown strained. And time pressures conspire to undermine thoughtful analyses. Taken together, these developments represent a sea change which has eclipsed the management methods that seemed to work well as recently as the 1990s.

The missing link

Yet despite their slide toward chaos, there's been no letting go of the necessity for getting product requirements right. If anything, the need to clearly define the elements of innovation, and to make sure the company's products include the right features, is more important today than ever – both for new products and for new releases of existing ones. However for most product managers, just keeping abreast of the surge in product requirements has become a full-time job. As a result, their accountability for leading innovation gets derailed. Their writing of requirements suffers. So does their ability to prioritize them.

They're not alone; that same confusion cascades down to the development teams charged with bringing those requirements to life. Tools which were once perfectly adequate are simply not up to the task today. Different stakeholders find themselves with different versions of the same spreadsheet – or sometimes without any version at all. There's no transparency. And nobody's sure who has the authoritative answer.

Nowhere is this disconnect more acute than in the markets for business software and software-driven devices. So it's no surprise that technologies have emerged to help beleaguered product managers deal with the onslaught. Different types of Requirements Management tools are now available from more than 15 vendor companies. But even with today's powerful new technologies, defining and managing requirements remains a huge challenge. A recent Forrester report [1] likened it to changing the tires on a moving car.

What has been missing until recently is a common plan of record – a comprehensive single repository of requirements, assignments, analyses and project status, accessible to everyone, that enables coordinated, end-to-end execution by the different players involved. The transparency of having everything in one database for all to see helps to open conversations and enable collaboration among participants. And because everyone is working from the same file of information, product managers, who no longer need to spend time chasing down dispersed data, have the breathing room to create clearer, better, tighter requirements.

Open collaboration also gives development team engineers a market perspective they might not otherwise see. They need to understand the What, Why, and When behind the features they're being asked to create. And they need to see the Who – to be able to trace the origin of a given requirement to its source, who may have a very specific need. Transparency allows the engineering team to understand who's asking for it and why. Without it, they could easily end up working on the wrong set of requirements.

The good, the bad, and the transparent

In an ideal world, detailed information about every aspect of the development process – custom-tailored to each participant's role – would be visible at a glance to all of its stakeholders. Status reports, discussion forums, roadmaps, assignments, ideas and analyses would all be immediately available to promote collaboration, track execution, and ensure customer connection. And, of course, requirement definitions would be crystal clear, notwithstanding the fact that clarity can be a challenge when those requirements grow out of broadly inclusive conversations in the marketplace.

Well-written product requirements are always a huge help to the company's implementation group. Even though their priorities are likely to shift in a dynamic Agile environment, their relative importance at any given time would provide valuable guidance to developers. As a result, engineers could keep their efforts constantly focused on the project's most important features. And they would have confidence that their work remained closely aligned with the company's most critical market drivers. Regrettably, however, well-written requirements and timely communications are not the norm in our less-than-perfect world. Untimely and badly written ones are far more common.

That can happen in many ways. One is missing requirements – specifications that arrive incomplete. Another is lack of clarity. For example, specifying that the software should be compatible with a particular platform, but failing to indicate which version of that platform it applied to. Conflicting requirements can become a huge headache. So can the failure to include explicit no-no's – features that the customer or market would find objectionable. And, of course, there's the failure to adequately prioritize requirements. Among the consequences: schedule delays, wasted resources, customer dissatisfaction, and internal strife.

Reference:

[1] "Right Tools. Write Requirements. Right On!" by Mary Gerush. Forrester Research. June 24, 2010.

Yet despite their surface differences, the root causes of most incomplete and poorly crafted requirements are actually the same. The lack of transparency – the inability of customers, product managers, developers and other key stakeholders to view the same information, to ask their own questions, and to see the full responses – triggers a flood of problems. It routinely leads to overlooked opportunities, lost insights, and needless misunderstandings among people whose interests and livelihoods require them to work well together. Overcoming that lack of transparency and getting everyone, literally, onto the same page provides a powerful antidote to the requirements chaos.

The lack of transparency - the inability of customers, product managers, developers and other key stakeholders to view the same information, to ask their own questions, and to see the full responses - triggers a flood of problems.

Pulling it all together

Connecting diverse stakeholders so that everyone has access to the same information at the same time is the essence of transparency. Enabling those individuals to work collaboratively toward the same goal through open conversation, and to apply their input into usable forms – even when what they’re saying is not what you’d like to hear – is its ultimate objective. But creating the opportunity for that to happen entails several foundational steps.

One is to build a central shared repository for ideas, suggestions, requirements, and the comments associated with them – a single, definitive source of truth into which everyone involved can see, evaluate, and use to move the right things into the marketplace.

The second is to nudge the organization toward a more Agile development process – one that values conversation over documentation as the preferred route to identifying ways of satisfying customer needs and getting higher quality products out faster.

Payoff is huge:

- Healthier Bottom Line
- Fewer Misses
- Better Products
- Faster Time to Market
- Better Strategic Fit
- Lower Development Costs

The third is to employ analytics that can prioritize requirements not only by their value to customers, but also by their fit with the company’s own strategic goals.

Each of those steps involves providing stakeholders, both in and outside the organization, with access to as much information as possible. Although the nature of their collaboration and the profiles of those involved will change as the project moves from one developmental phase to another, it means integrating methodologies and tools into the process that would allow those participants to work together effectively through every stage of the company’s product lifecycle.

Those are major changes, but their payoff is huge: a healthier bottom line, fewer misses, better products, faster time to market, better strategic fit, and lower development costs. And along the way, it enables product managers to keep a highly complex process from slipping out of their grasp.

You can do it, too

Accept Software delivers The Product Innovation Management solution that technology-driven companies use to decide which products to bring to market for the fastest time-to-profit. The Accept360 Suite is the only complete end-to-end software solution that tightly links company and product strategy through ideation, portfolio planning, roadmaps, requirements and execution. For more information, visit www.accept360.com or call us at 512-201-8222.

About Accept Software Inc.

Accept Software, Inc. delivers The Product Innovation Management solution that technology-driven companies use to decide which products to bring to market for the fastest time-to-profit. The Accept360 Suite is the only complete end-to-end software solution that tightly links company and product strategy through ideation, portfolio planning, roadmaps, requirements and execution. For more information, visit www.accept360.com

401 Congress Ave, Suite 2650,
Austin, TX 78701
512-201-8222

www.accept360.com
info@accept360.com