

Behavioral Patterns at the Heart of Cloud Software Design

Software developers have taken their users for granted for too long. This includes users inside and outside of the procurement function, and this realization mirrors changes being seen in the industry's priorities and objectives.

In a savings and mandate-driven environment, business software did not need to compete with the visual appeal and usability of consumer software. Now, with business users expecting the intuitive interfaces and easy-to-use functionality available in consumer apps from their business software, it's imperative that business software developers place the user at the center of the design. The changes required are not incremental; it is time for a full paradigm shift in business software design.

This white paper discusses the behavioral drivers behind the paradigm shift required in business software design for greater user adoption, as well as how those behaviors translate into the technical requirements and usability expectations of the next generation of solutions.

Software has two basic functions: either to assist humans in completing tasks or to replace their effort through the automation of those tasks. Today, enterprises need both of these functions to be met by their software. Unfortunately, many solutions in the marketplace are not a match for current user behaviors and automation priorities.

Aligning Business Software with Existing Workflows

Although any software enables a distinct set of tasks, designers must remember that after implementation it must also function as part of a larger end-user ecosystem. Within that ecosystem, there are multiple types of users, tasks, and processes. While process automation incorporates all three factors and ultimately eliminates the need for human intervention, workflow automation (the type of automation most commonly seen in procurement software) enables tasks and processes, while leaving users in place to continue their work through intuitive software that minimizes the time required for manual tasks.

Making sure that business software aligns with existing workflows requires a careful study of the tasks and thought processes of users. Those users must be placed at the center of the design.

Making sure that business software aligns with existing workflows requires a careful study of the tasks and thought processes of users. Those users must be placed at the center of the design. While this seems obvious, this actually requires a radical change in designer mindset. Historically, manual tasks and processes were automatically considered 'broken.' The software designed to 'fix' them showed no regard for the reasons behind those processes or for the users that carried them out.

The result may have been an improved, more automated business process, but it came with significant costs and change management challenges that doomed its hopes for acceptance from the outset.

Today's thinking on the subject sets a much higher bar for designers. They must understand how users work in their native environment and use software to improve and simplify – not change or fix – existing processes. Software must identify and accommodate the complexity faced by users and then provide a solution to their challenges that they will willingly adopt.

Users expect to see intelligence in their software. They want it to be responsive, flexible, and context sensitive. They also want to see evidence of 'forgiveness' – a human trait that significantly improves usability and satisfaction by accommodating the naturally occurring irregularities of real life business management.

In many cases, intelligence is exhibited through software that is able to 'learn.' Although that capability feels more like the stuff of artificial intelligence (AI) than business software, there are some straightforward applications. For instance, it is possible to leverage pattern recognition techniques to bring the most frequently used tasks to the fore. The alternative to this is a rigid navigation path that requires all users to follow a predetermined step-by-step process that includes every possible option, regardless of how frequently they are applicable to that user.

Keep it Light and Simple

Part of how business software can be flexible for every user in every process and in every context is by keeping the platform as agile as possible. 'Light' designs take complex processes and support them through powerful but minimalistic software.

A visually minimalistic user design requires enough white space to create a clean UI. The clean workspace is created and preserved by starting simple and then continuing in that spirit, prompting users with the

option to hide never-used features or only offering up functionality once user inputs indicate that it is relevant to that task or context.

Less is also more when it comes to putting banner messages and alerts in front of users. People naturally become blinded to repetitive messages – even if they offer up useful or important information. Rather than text, visual cues provide an effective way to communicate alerts and confirmations. Ideally, users establish a two-way ‘dialogue’ with the software. Unlike the task-driven software of the past, user-centric business software is process- or workflow-driven, and never forgets that there are larger forces at play than the immediate series of clicks and button pushes.

When software requires many steps and significant interpretation from its users, there is a corresponding increase in the amount of training required to make it usable. Training is costly, disruptive, and rarely as effective as planned. Ideally, enterprises should select software that doesn’t require extensive end-user training. To make this possible, the users themselves must be well-understood and their requirements must be at the center of the design before the software is presented to them.

For instance, software must ‘understand’ what kind of tasks users will perform on what kind of devices. Most users still want to complete heavy-lifting tasks on desktops and laptops, while they are comfortable doing light work, such as viewing dashboards or reviewing approval requests on phones and tablets. These preferences must be incorporated into multi-device designs while preserving the consistency of function and appearance.

Just as important as understanding how users will connect with the software is understanding when and where they must transition in and out of it. These are precarious moments from a usage standpoint. If handled well, they can ensure adoption success, and if handled poorly, they may give users a reason to attempt to bypass the software altogether. Similarly, software cannot assume that usage means one person in front of one screen; users do not function in isolation. Software must support virtual collaboration through, perhaps, long distance screen sharing, as conveniently as it provides information directly to one immediate user.

Unlike the task-driven software of the past, user-centric business software is process- or workflow-driven, and never forgets that there are larger forces at play than the immediate series of clicks and button pushes.

Interactive Design

When it comes to incorporating visual design elements and facilitating interaction between humans and computers, there are conventions at play that most users are not consciously aware of – nor should they need to be.

Although we use our eyes to read as well as to look at graphics, most people naturally find it more helpful to rely on visual cues than to read text. This plays out in software design through the distinction between recognition and recall. Recall requires users to read to find meaning. Recognition allows users to instantly make a choice or decision based on remembered meaning, alleviating the need to read and speeding up the interactive process.

Recognition is not possible without consistency, both within business software and beyond it. Some conventions are nearly universal and have been established over time. Some – such as the icon of a floppy disk for ‘Save’ – have even outlived the roots of their meaning but are so pervasive that they endure nonetheless.

These ‘metaphors’ – or visual cues that have consistent meaning across platforms and software applications – contribute to feelings of familiarity between users and software. Scrollbars need to look familiar and be placed in expected locations. Draggable elements must ‘look draggable’ rather than being labeled as such or requiring training and memorization.

In addition to visual cues, gestures should take full advantage of how software can be used through a range of interfaces on a variety of devices. Matching users with gestures and tasks will not only help them navigate intuitively, but make usage more memorable. Drag and drop, swipe, double click, right click, and pinch are all options, and gestures across devices should correspond. For instance, a mouse click on a laptop should be replaced by a finger tap on a phone or tablet. The task is the same, as is the user in many cases, and they instinctively know how to interact through previous experience using platforms across devices.

Recall requires users to read to find meaning. Recognition allows users to instantly make a choice or decision based on remembered meaning, alleviating the need to read and speeding up the interactive process.

Users interact with software through their eyes, hands, and minds. All three interaction types need to be supported across a range of devices, and tap into familiar conventions – whether visual metaphors or interactive gestures – to create feelings of understanding and alignment toward the software.

Conclusion

In the design process for business software, there must be both general and business-specific elements. Investments made in research about the actual usage patterns of users ensure that the software designers’ intent is carried out in practice.

As changes are introduced, whether due to technical advancements or shifting demand from an evolving user base, they must be evaluated for alignment, simplicity and consistency to ensure that they further engage – rather than distance – each user group from the software put in place to assist them.

Key Take-Aways:

- The ‘flow’ is more important than the ‘work’ in workflow. Users want to be assisted by software, not corrected by it. All automation must approach current user practices with a tone of forgiveness so that the software designed as a result can do the same.
- Simplicity may belie a great deal of complexity. Software does not need to look complex to be complex. In fact, the most capable software enables users by managing the display of functionality for them through pattern-based prompts and gradual disclosure.
- The wheel doesn’t need to be reinvented; just fit to each unique purpose. Many of the key visual cues and interactive gestures that make software easy to adopt have been established by other solutions and operating systems. By incorporating them into business software, successful designers can both increase adoption rates and minimize the need for investment in training and change management. ■

Recommended reading for sourcing and procurement professionals

Procurement Software GPS: Strategies To Navigate The Technology Landscape

In this new white paper, noted industry veteran Andrew Bartolini of Ardent Partners and experts from GEP highlight strategies to help you find the right procurement software for your enterprise. The paper also shares useful tips and advice to help procurement teams gain maximum value from their current technology investments. [DOWNLOAD NOW](#)

Source-to-Pay Automation Playbook

This new white paper from Procurement Leaders Magazine shares authoritative insights and practical advice from leading S2P technology experts on what to do AND not do while implementing S2P Automation. An essential resource for procurement pros looking to drive higher savings and performance from source-to-pay software. [DOWNLOAD NOW](#)

More on Cloud and Mobile Technology

For more white papers, case studies, webcasts and other resources on cloud and mobile technologies for procurement professionals, please visit the GEP Knowledge Bank.



To learn more about GEP's consulting and outsourcing services please visit www.gep.com



To learn more about cloud and mobile procurement software please visit www.smartbygep.com

Access our free knowledge resources at <http://www.gep.com/knowledge-bank>

100 Walnut Avenue, Clark, NJ 07066 | P 732.382.6565 | info@gep.com | www.gep.com

Clark, NJ | Princeton, NJ | London | Prague | Mumbai | Hyderabad | Shanghai | Mexico City | Costa Rica | São Paulo | Singapore | Sydney
Copyright © 2016 GEP. All Rights Reserved.