

On The Road To Agile Mobile Business Applications

Consumer-level applications are a catalyst for mobile business applications.

The proliferation of consumer-level applications on mobile devices (e.g. iPhone, iPad, Android etc.) is impacting the business world. Business people expect to have the same connectivity and ease of use with their company's business applications on their mobile devices as they do with consumer applications, such as travel instructions or making reservations in a restaurant. They also expect a far higher speed of delivering such business applications. Why should a mobile business application take years to develop, given the rapid appearance and updates of consumer apps?

There Are No Simple Mobile Business Applications

Once you go beyond the most trivial translations of simple business processes (e.g. retrieve price book), you find that there are no "simple" mobile business applications. Regardless of how fast you wish to move to actual development, you have to start with some minimal scoping of the application and its boundaries, understanding the interplay of the same application on mobile devices of several roles (e.g. a subordinate and their manager in a vacation-approval workflow), the workflows on mobile devices and their interactions with workflows on the back office systems, and more. Prior to going live, quality assurance, security, and scalability testing are also steps that should not be underestimated — thus, the need for a platform for agile development, deployment, and ongoing evolution of mobile business applications.

The need for agility rules out the traditional software development process. Not only is this process far too slow and expensive, it often produces solutions which, no matter how perfectly they match the documented requirements, still fail to gain user acceptance in the real world. Why does this happen? Veterans of such projects know you cannot expect a business user to read a huge document and translate it to the way the solution will work for all the varieties of a real-life situation.

Based on experience in developing and deploying mobile workforce management and service optimization solutions, we formed the WISIK paradigm to answer this dilemma: "When I See (it) I'll Know (it)." The sooner the user sees a solution and tries it out, the sooner they can point out what they like and what they want changed. WISIK's premise is:

1. It is easier to modify a full working solution than to build it from building blocks or from scratch.
2. Watching a working product in a variety of scenarios is faster and better for grasping what it could do — the product is the best documentation!
3. Only doing "no-coding" configuration changes can shave off several weeks of testing and can quickly cycle through user comments and requests.

Having a starter kit is key for the WISIK methodology. You start with a working solution and configure, customize, or extend it so that a working solution is maintained, and you can examine its behavior with the intended business users. At a certain point, the business users reach the WISIK point, and then you are ready to package and test the application.

The "no-coding" point is a crucial element of this paradigm. For instance, of the 90% of the apps required for the field service sector, you can fulfill 90%+ of the requirements by no-coding configuration if your starting point is a well-designed, mature product with proven field experience. When you do no-coding configuration, the likelihood of introducing new bugs is minimal, since the testing has already been done during the product's Q&A process and covers a significant portion of testing.

The key success factor for a WISIK-based development and deployment platform for a set of industry verticals is having the domain expertise and a solution that is close enough to the required end result, while possessing substantial expressive power to close any remaining gaps with minimal coding. If this is satisfied, then you are already on your way to develop and deploy mobile business applications almost as rapidly as consumer-level mobile applications. The key metric for comparing development platforms is: How far can you fulfill the application's requirements by no-coding configuration before you resort to coding a requirement in a programming language? •



Dr. Moshe BenBassat
chairman and CEO,
ClickSoftware,
www.clicksoftware.com

moshe benbassat