

MobiCASE 2010
Workshop on Mobile Software Engineering
Position Paper Template

Event: October 28, 2010.
Mail paper to wmse@sv.cmu.edu by July 15, 2010.
No more than 2 pages, 11 point font.

Paper title: Mobile Software Engineering: it's just a domain
Name: *Todd Sedano* Role/Title: Director of Software Engineering
Organization: *Carnegie Mellon University* Preferred Contact: todd.sedano@sv.cmu.edu

Please state your point of view on the issues of Mobile Software Engineering. Please specifically address the following 5 questions:

1. Your experience as a software engineer?

I'm a practicing software professional with fifteen years of experience writing and delivering software systems. For the past ten years, I've worked mostly with startups.

2. Your experience with mobile software development?

I've co-taught courses on mobile software development as well as mentored student teams on mobile software development.

3. How does traditional software engineering relate to the engineering of mobile applications and systems?

From my perspective, mobile software development should employ the same methods and processes as traditional software development. Yes, there are unique constraints in the mobile domain. Just like any other domain, the mobile domain has specific techniques that are relevant for it. For example, when we look at SWEBOK, most of the practices are relevant for the mobile domain. When we look at GSwE 2009, mobile is a great domain to study software engineering fundamentals. Interesting several mobile developers have commented that mobile development is more lot like web development, than packaged software development. In packaged software development, there is limited and carefully controlled release, whereas with mobile and web development, the product is updated frequently. Companies treat their web site and each mobile the same, they are just another application view.

Mobile development does have some unique challenges.

-- Mobile devices are harder to test than say web applications or desktop software, but no harder to test than other embedded systems. In fact, their proliferation makes it easier than other embedded devices since there are a number of testing services that will test one's mobile application on different platforms and provide screen shots of error conditions. (See Device Anywhere's offering)

-- Testing the application should include testing under typical limited resource conditions. Developers need to purposefully test the application under low memory conditions, and cover the phone with aluminum foil to see how the app responds when it loses packets and the network.

-- the number of platforms and devices continues to be a huge headache. With Apple's requirement of a dataplan and the advent of the iPhone store, developers can now develop for a single platform since the market is large enough to support single platform development. One recent study showed the overlap of the iPhone and Android marketplace is only 2%.

-- Finally, business differences: customers quickly evaluate and have little tolerance for usability issues (hci testing), revenue models can be tricky given low prices of apps

4. What are the distinguishing features of mobile software specification, architecture, development and testing that need special attention, skills, or innovation?

-- hardware is innovating very quickly. Even on the iPhone platform, which releases once a year, whenever the new OS comes out, development teams need to drop everything, put their plans on hold, and upgrade their products. Because customers upgrade quickly, they expect their applications to continue to work. Often there is backwards compatibility issues. Furthermore, Apple features products on the current OS.

-- memory management (although this is true for most desktop software)

-- even though the cell phone is a resource constrained device, many developers really don't worry about power consumption or difference in speed of CPUs (unless they are working on a game.)

5. What is the suggested focus and agenda for mobile software engineering research and education?

-- human computer interaction including touch screen interface metaphors

-- simplifying testing on all devices,

-- creating uber simulators that run user inputs on multiple simulators at once,

-- any framework aids to deal with quickly changing technology