

Guest Post: The Past, Present and Future of Apphance

Posted on 08/17/2012 in [Guest Posts](#), [uTest Stuff](#) by [Guest Blogger](#)



As you've probably seen by now, uTest acquired Apphance in a seven-figure deal. You can get more details about the deal in this [blog post from uTest CEO Doron Reuveni](#). You can also read more about the product itself in this [subsequent post from Stanton Champion](#). But in this post, we wanted to share the origins of Apphance from Jakub Lipiński, the CEO and co-founder of Polidea, Apphance's parent company. Enjoy!



In order to diagnose what's gone wrong with mobile apps, developers need specific information. The problem is that this information is extremely hard to find and even harder to communicate.

Consider the following conversation (*sadly, based on an actual call*):

Developer: I've got a bug saying that the app doesn't look good on your device. Could you give me a little more information about what kind of phone you have please?
Customer: It's white.
D: I actually meant what's the model?
C: Aaaah! Ummm, it's an HTC.
D: Ok, what kind of HTC?
C: Ummm...Desire.
D: Cool, and what version of the firmware are you running?
C: Firmware? What do you mean? Is that like hardware?
D: Hmmm, never mind. Could you send me a screenshot of the problem?
C: Okay... sure, but I'm not quite sure how to do it...
D: Let me explain: Install Java JDK. Install Android OEM USB Drivers. Install Android SDK. Reboot your computer. Connect the phone to the desktop. Launch the Dalvik debugger. Go to Device->Screen capture. Make the screenshot. Save it to disk. Compose a new email and attach the file. It's that easy.
C: ...Nevermind, it's fine.

The point: devs and users rarely speak the same language. This causes misunderstanding and frustration. It also prevents developers from improving their mobile apps and rarely closes the loop between users and engineers.

This brings me to the origins of Apphance. At Polidea, we envisioned a mobile quality tool that would gather instant (and accurate) information from testers and end users about their app experience. We wanted it to be fully automated; to provide the necessary information developers need to diagnose the issues and allow testers to clearly indicate what they would like to improve. We scoured the mobile tools landscape for this type of functionality and came up empty.

And after several months of fruitless searching, we went ahead and created a suite of embedded software libraries that solved our own pain points. These libraries automatically send all the information about the device (maker, model, OS and version) to the server where the developers can view them.

Whenever external conditions are changed during the test (connection status, screen orientation, battery life, etc), it's also sent to the server. If the app crashes, the developers know about it right away, as well as a mountain of valuable diagnostic information to help them find the root cause. And if testers want to report a problem they encountered while testing the app, they just shake the device. A screenshot is captured; they can draw on the screenshot to highlight the issue and provide additional commentary. The report goes straight to the people responsible for the app, and it can also be submitted to the company's bug tracking system.

As we rolled this tool out to our mobile customers, two things became clear immediately:

- We weren't the only ones feeling this pain point.** As an entrepreneur, when you identify a gap in the market, there's always that nagging doubt: "Maybe I'm the only one who sees this as a problem ... maybe I'm fooling myself into thinking this is a big deal." It quickly became clear that the vast majority of mobile app developers & testers felt this same pain.
- Devs, testers and users loved it.** Using Apphance during development smoothed the communication between developers, testers and users. Developers and testers loved it because they finally had accurate, diagnostic information from users. Users loved it because it was so easy for them to provide precise feedback regarding the application progress and describe the changes they would like us to introduce.

We soon realized that Apphance might also be helpful for other mobile app developers. So we spun it off from Polidea and took it to market as a standalone entity. The feedback from the market was very positive and Apphance started to get some traction.

At the same time, we started looking for a partner who could help us make the tool much more popular. And uTest was the first company on our short list. They already had relationships with all our potential customers, had a bigger brand footprint and a proven distribution channel. Plus, Apphance was a perfect complement to their mobile testing offering. We first met their CMO, Matt Johnston, at Mobile App World in London in mid-2011. We followed up in Barcelona at Mobile World Congress, and ran into them again at SXSW in Austin. Then we visited uTest headquarters in Boston during April of this year. We initially discussed partnership and integrations, but then mutually realized that closer collaboration made the most sense.

And the rest, as you know, is [history...](#)

So not only are we wildly excited about the future of Apphance and its integration with uTest, but we're also excited to overhear the following conversation in the not-too-distant future:

Developer: I've noticed there was some issue with our mobile app. I know you haven't reported it yet, but we wanted you to know that it's already been fixed :)

To close, I wanted to thank the hard-working staff at Polidea, as well as the early adopters of Apphance for your continued support and feedback. We couldn't have done this without you. The best of Apphance is yet to come, so stay tuned!

One Response to "Guest Post: The Past, Present and Future of Apphance"

[Five Blogs – 20 Augustus 2012](#) « [5blogs](#) said:

[...] Guest Post: The Past, Present and Future of Apphance Written by: [Guest Blogger](#) [...]

Posted on [08/19/2012 at 9:21 pm](#)

Leave a Reply

Name (required)
 Mail (will not be published) (required)
 Website



Categories

- Agile Dev & Testing
- Crowdsourcing
- Guest Posts
- Product Updates
- Security Testing
- Start-Up Stuff
- Tester Community
- Testing – Mobile Apps
- Testing – Web Apps
- Testing the Limits
- Testing Trends
- uTest Stuff



Most Popular

- 6 Testing Mistakes to Avoid Like the Plague
- Testing the Limits With Jane Fraser
- Why the Testing Experts are Right (Part 1)
- Ruby on Rails Security Vulnerability Throws Apps Off Track
- 6 Things You Should Never Say to a Software Tester
- When Should You Start the Testing Process?
- Testing the Limits With eBay's Jon Bach – Part I
- 16 Great Quotes For Software Testers
- Forbes Names uTest 8th Most Promising Company in America
- Does Being a Video Game Tester Actually Suck?

