Editor: **Robert Blumen**
SalesForce
robert@robertblumen.com

Matt Lacey on Mobile App Usability

Gavin Henry

From the Editor
Matt Lacey, author of *Usability Matters*, discusses usability for consumers and business or in-house users. Host Gavin Henry spoke with Lacey about the six components of great app experiences, things every app should do, native apps, password managers, accessibility, feedback, telemetry, locations, nonmobile devices, examples of good and bad apps, testing, connectivity, user involvement during development, and usability and software engineering. We provide summary excerpts below; to hear the full interview, visit http://www.se-radio.net or access our archives via RSS at http://feeds.feedburner.com/se-radio.—*Robert Blumen*

Gavin Henry: What is usability?

Matt Lacey: It means software is simple and does everything the person using it needs it to do without trouble.

Why is usability important?

For a consumer app, customers have alternatives. For a business app used internally, the experience of using it affects productivity and how people feel about their work. More attention is usually paid to consumer-facing apps than to internal apps. But internal software affects productivity, morale, and the ability to work remotely.

Digital Object Identifier 10.1109/MS.2020.3042424
Date of current version: 11 February 2021

Is usability the same as an experience?

An experience is what it is like to use the app. That's usability—how does the person using it feel, and how productive are they? Can they do what they want and achieve their goals? An app can be perfectly functional but horrible to use. Functionality is not the same as usability.

What are the six components of great app experiences covered in your book?

Context, input, output, responsiveness, connectivity, and resources. These are things beyond code that are important and easy to overlook.

Context is understanding where the app is used in addition to how it actually works: who uses it, what they want to do with it, how they use it, what devices they use it on, where they are located, what languages they speak, and so on.

Input is how data get into the mobile app, both from the person using it and from other sources. Information from sensors on a device or from a remote source is input in addition to what gets typed in through the keyboard, and you need to think about it.

Output is what I put on the screen and more, such as sound and sending emails. And where will those emails be received? On the same device as the app? I've seen apps where you sign up through the app and it sends you an email, and then you can't view the app and the email easily at the same time on the device.

The fourth component is *responsiveness*. How do we respond, or how is our response perceived? Do users get a quick response when they click a button? Did the mobile app immediately start doing something, or did it sit and spin and make the user wonder if it was all okay? How does the user feel about the speed of the response? You have to make sure that things happen fast enough for the person using it. It comes back to context.

Must the developer know the context up front?

I always like to know context, to be sure I'm building something that's appropriate for users. We can measure, test, and ask for feedback during development. I do user-based testing, getting software in front of real users as soon as possible. The feedback I need varies depending on the application, but I make sure that people can provide feedback and that the software is being used by users in the real world, as much and as soon as possible. In those early stages, there can't be too much feedback before you get to the App Store. Once you go public, there will be a lot of new feedback coming—the real world can always find new things for you.

Fifth is *connectivity*. Not everyone always has a high-speed Internet connection; being occasionally connected is the norm. You have to think about whether you can do without connectivity, what to do when it goes down, and how you can work around it not being there or if it keeps dropping and coming back—how do you live with those scenarios? To provide a good experience using an app, you should always make sure the user knows what's going on. They may think, *Did it send? I'm offline, what's happened, what does that mean?* When they're asking questions like these, their experience has been negatively affected. Uncertainty in the user is always bad.

Sixth are *resources*. The biggest resource you need to consider on a mobile device is power—will what I'm doing drain the battery? No one will use your app if it makes their battery run down quickly. The next biggest one is disk space. Mobile phones are highly constrained in terms of the space they can use. The availability of disk space for users who have many images on their phones might be something you need to consider.

Can you use telemetry data to monitor and understand your app's battery usage or storage needs? Or do premium phones on modern operating systems already limit battery usage by apps anyway?

Measuring power consumption or how quickly you drain the battery can be harder to test for than other things. The operating system can take care of some of that, and there are settings reports on how much of the bandwidth, power, and screen time the app is using. But you can't rely on the operating system to take care of this for you. The operating system will do what's best for the operating system, but you need to do what's best for your app and your users.

You don't need telemetry for disk space, but you can check how much disk space is there before trying to save images locally,

SOFTWARE ENGINEERING RADIO

Visit www.se-radio.net to listen to these and other insightful hour-long podcasts.

RECENT EPISODES

- 437—Tim Sneath, product manager for Flutter and Dart at Google, discusses with host Gavin Hendry what Flutter is, why it was created, and where Dart came from, as well as what the different layers of Flutter are and why it's so popular.
- 436—Yi Pan, lead maintainer of Apache Samza, discusses the internals of the Samza project as well as the Stream Processing ecosystem with host Adam Conrad.
- 435—Julie Lerman discusses Object–Relational Mappers and Entity Framework with host Jeremy Jung.

UPCOMING EPISODES

- Host Justin Beyer discusses cyprography with Jean-Philippe Aumonsson.
- James Smith talks with host Priyanka Raghavan about software bugs.
- Host Kanchan Shringi discusses UX for Enterprise Software with Arin Bhowmick.

save data, or make a copy of the database, which will take up a lot of space. You have to be aware of the resources you're using and be sensitive to the needs of the device as well as the needs of your app. Resources also include things such as the device camera—how many, and how do I use them?—and the location sensor.

What is one important thing that every app should do?

Basic analytics and error reporting—if something goes wrong, you want to know about it. Don't rely on the user telling you because they won't know enough to give you the information you need. Analytics tell you what devices and operating systems people are using. You might want to use this new feature, but it's available only in the latest version of the operating system. If your users don't update to the latest operating system quickly, then that's a waste of your effort.

ABOUT THE AUTHOR

GAVIN HENRY is the founder and managing director of SureVoIP, an Internet telephony service in the U.K. Contact him at ghenry@surevoip.co.uk.

IEEE Software (ISSN 0740-7459) is published bimonthly by the IEEE Computer Society. IEEE headquarters: Three Park Ave., 17th Floor, New York, NY 10016-5997. IEEE Computer Society Publications Office: 10662 Los Vaqueros Cir., Los Alamitos, CA 90720; +1 714 821 8380; fax +1 714 821 4010. IEEE Computer Society headquarters: 2001 L St., Ste. 700, Washington, DC 20036. Subscribe to *IEEE Software* by visiting www.computer.org/software.

Postmaster: Send undelivered copies and address changes to *IEEE Software*, Membership Processing Dept., IEEE Service Center, 445 Hoes Lane, Piscataway, NJ 08854-4141. Periodicals Postage Paid at New York, NY, and at additional mailing offices. Canadian GST #125634188. Canada Post Publications Mail Agreement Number 40013885. Return undeliverable Canadian addresses to PO Box 122, Niagara Falls, ON L2E 6S8, Canada. Printed in the USA.