

Ten things you might want to know before building for mobile

Ken Banks & Joel Selanikio

Progress in the social mobile field will come only when we think more about best design practices rather than obsessing over details on the ground. Social mobile tools are those built specifically for use by organisations working for positive social and environmental change, often in the developing world. Over years of creating some of the most widely used mobile applications in the public space, we've made a lot of mistakes, and we've learned a lot. We think that successful mobile projects – those aimed at developing countries in particular – have a better chance of succeeding if these points are considered from the outset:

1 You will never know what the end-user knows

All the best technologies – from fire to phones to cars to writing to email – all of them are general purpose solutions that solve one problem – transport, cooking, communications, etc. – in general but not in particular. That is because there are too many particular, on-the-ground situations – too many things to write about, too many things to talk about, too many places to drive to – for the technologies to ever anticipate them all.

So don't try: make it your goal to design the spreadsheet, the email, the general tools so that users – who know their own needs better than you ever could – can repurpose them to suit those needs. That approach lets users create their own solutions, using your tools, and creates a sense of local ownership, which is crucial for success and sustainability. It's always going to be easier to equip local NGOs,

or users, with tools to do the job than it will ever be for you to learn everything they know.

2 Aim for the technologies most widely available to your users

Ensure that your applications can work on the most readily available hardware and network infrastructure available to the user group you're aiming at. Text messaging solutions aren't big in the social mobile space for nothing: they're simple, and they're available to anyone with a phone. If your target audience is the rural public in Africa, a Web 2.0 application wouldn't make a lot of sense.

On the other hand, it also wouldn't make sense to restrict political workers in Eastern Europe from using a web-based application. So consider your users and if in doubt go for the simplest platform first.

3 Don't reinvent the wheel

Check to see if any similar tools to the one you want to build already exist and, if they do, consider adding to them rather than starting from scratch. People and institutions are incentivised to reinvent the wheel each time, but don't do it unless you really believe there's nothing out there you can use.

4 Simple and free scales better than complicated and expensive

Anything that needs a programmer or technologist to use is inherently less scalable

than something (like the car, like the phone, like email) that can be used by the average non-technical user. So from the outset try to build something that's easy enough to use without the need for user training or a complex manual (or any manual at all!) – so new users can easily and effortlessly replicate once news of your application begins to spread.

Be realistic about what your application can achieve, and wherever possible look for low-hanging fruit. Remember – big is not better, small is beautiful, and focus is king. A solid application that solves one element of a wider problem well is better than an average application that tries to solve everything (especially given point 1, above).

Another factor in keeping it simple is remembering that every third party the user needs to speak to in order to implement your solution increases the chances of failure by a considerable margin, particularly if one of those parties is a local mobile operator or a high-priced foreign consultant.

5 Focus first on the users, not the developers

Anyone who builds software inevitably spends more time with developers than with users – especially if your users are in some of the more hard to reach spots on earth. Don't let yourself get sidetracked by technical details that the user doesn't care about but that developers love to discuss; that's as silly and time-wasting as arguing about Windows vs. Mac. The user cares about cost, and the user cares about simplicity, and the user cares about whether the software gets the job done. That's what you should care about, too.

The best example of this developer-focus is the constant discussion about open-source. Open-source is great for some things and not great for other things, but that's an issue you can deal with after you have working software: the start of a project is not the time for "design by committee" anyway. Controlling your development process to start with also helps you understand better who is using the app – something that donors routinely want to know. Besides, if you can give your users something as elegant, simple, and free as Gmail (free but closed source, like most widely-scaled and popular web applications) they will be very happy users.

And encourage those users to share experiences, and to support each other. Don't be afraid to reach out for additional information, and work hard to keep it active, engaging and growing. Solicit feedback, and criticism. Communities are notoriously hard to build, but when they work they're worth it.

6 "Shipping is an important feature"

This dictum of the best programming shops reminds us that good software in the hands of the user is always better than perfect software that no one ever sees. Think about rapid prototyping. Don't spend too much time waiting to build the perfect solution, but instead get something out there quickly and let reality shape it. Get user feedback. Then get more user feedback.

7 Promote your solution like crazy

Reach out to people working in the same technology circles as you, post messages on

relevant blogs, blog about it yourself, speak at user and developer conferences, build a project website, brand your solution, and make use of social networking tools such as Twitter and Facebook. Make your users aware, make your funders aware, make the developers aware, make the media aware.

8 9 & 10 Don't let anything stop you

Not a lack of funding: if considerable amounts of funding are required to even get a prototype together, then that's telling you something – your solution is probably overly complex.

Not a lack of specialists: nowadays it is easier than ever to learn programming, or to communicate to a worldwide audience. Learn to do what you can't afford to pay other people to do. The more design, coding, building, testing and outreach you can do yourself, the better. Stay lean. These tasks can be outsourced later if your solution gains traction and attracts funding. The more you achieve with few resources the more commitment and initiative is shown, increasing the chances a donor will be attracted to what you're doing.

Not the naysayers: many people will stand on the sidelines and tell you all the reasons why it just won't work. Ignore them. Those people don't build, they prevent building. Ignore them. Remember these words from the writer Arthur C. Clarke, and forge ahead:

New ideas pass through three periods:

- 1 It can't be done.
- 2 It probably can be done, but it's not worth doing.
- 3 I knew it was a good idea all along!



Ken Banks is the founder of kiwanja.net and the creator of FrontlineSMS, a piece of free and open source software which turns a laptop and mobile phone into a two-way group messaging hub. Ken combines over 25 years in IT with 16 years' experience living and working in Africa, and has a degree in Social Anthropology with Development Studies. He has been working exclusively in mobile for the past seven years. Ken recently became a Tech Award Laureate for his work with FrontlineSMS. www.kiwanja.net



Joel Selanikio is the co-founder of DataDyne.org and the creator of the widely used EpiSurveyor mobile data collection web/mobile app. A practising paediatrician, and winner of the Lemelson-MIT award for Sustainable Technology, Selanikio was named as one of the seven most powerful innovators by Forbes magazine – and he is daily amazed by the power of web and mobile to better our world. www.DataDyne.org