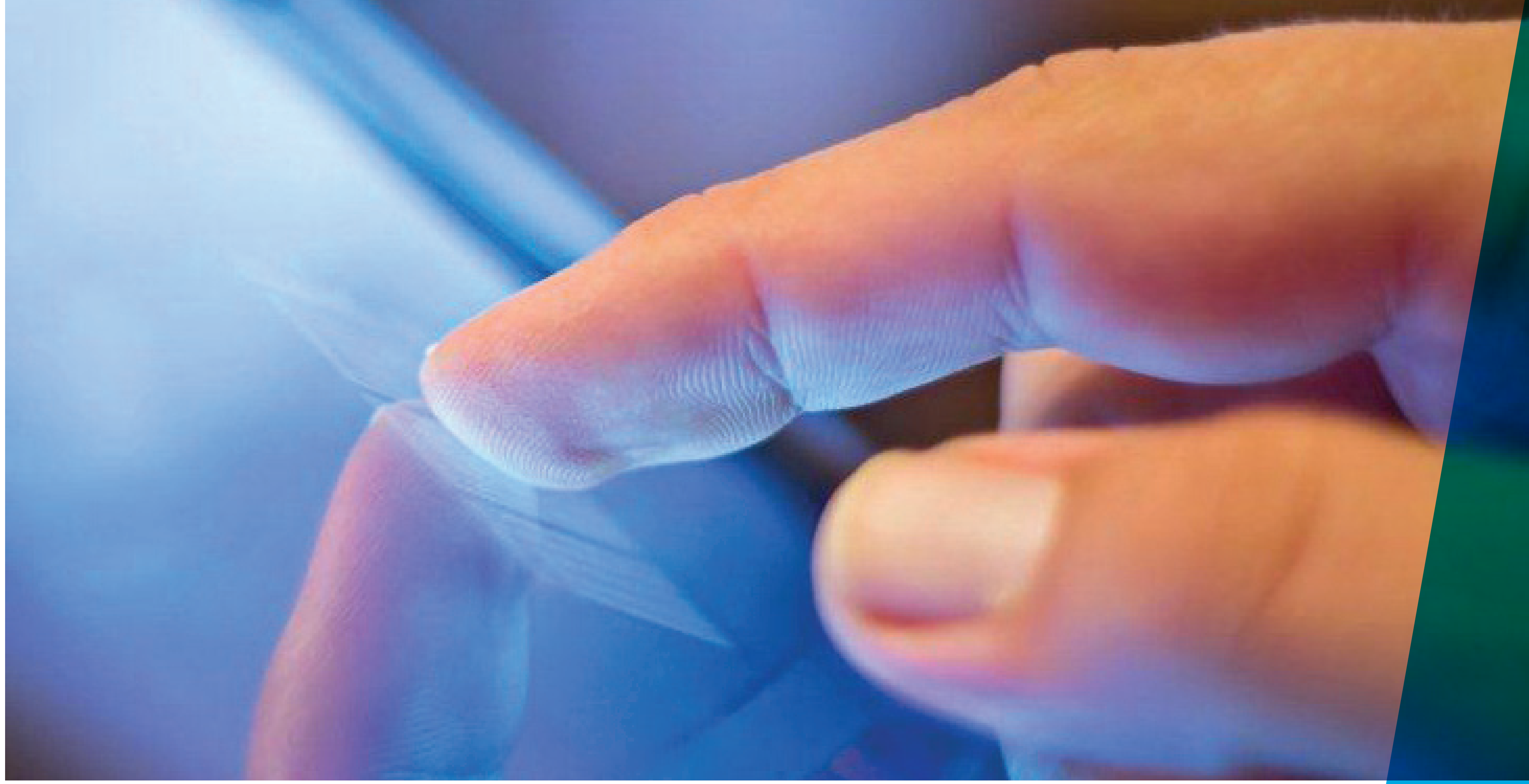
How to Avoid the Top 3 Gotchas of Mobile App Development

By: Sean Johnson of Field Data Integrators

3 The Gotchas of App Development



Why You Should Read This!

More and more managers are getting besieged by requests for custom mobile apps to automate their unique business processes and workflows. A big driver of mobility for many enterprises is the opportunity to improve customer and field service with instant data capture, access, and workflows.

App development using web-based forms for always-connected workers with PC browsers, for example, can be fairly straightforward. Challenges arise for teams needing fully customized apps that can be used while workers are online or offline. Despite general improvements in network coverage, many workers in the field don't have reliable, snappy coverage when they need to capture or send data.

To get around connectivity issues, teams usually build local apps on tablets and smartphones, which can be expensive and time consuming. A typical enterprise iPad app with medium levels of integration has an average development cost of $100,000. The vast majority of those apps require 3-12 months of development time.

With the expansion of the mobile development ecosystem over the past few years, there is now a wide variety of different development tools and options for building apps. There is also a lot of accumulated wisdom for helping teams avoid the common pitfalls or "gotchas" of mobile app development.

Here are the top three avoidable gotchas of mobile app development to ensure a successful roll out:

- Everyone likes your mobile app except the workers – and they don't use it
- Getting locked into proprietary solutions that stakeholders – such as IT – don't like
- Not preparing for change

About the Author

For the last 20 years, I've been helping oil + gas, utilities, telecommunication, local and federal government field teams build field data capture solutions for GIS, GPS and digital paper field service, inspection and maintenance reporting across a range a devices including tablets, laptops, smartphones, and digital pens.

Gotcha #1:
Everyone likes your mobile app, except the workers – and they don't use it

All too many teams have invested large amounts of money in mobile apps only to see them fail in the field because the teams don't like them. The source of the issue can range from the device itself, the app layout, the sequence of the workflow, or even the size of buttons.

The key to avoiding this gotcha ▸ get mobile worker feedback from the start.

It's critical to work with mobile teams to not only understand their workflows – but to also understand the environment in which they are working.

Some environments may be best suited to rugged devices, or even pen and paper. An app that is snappy and responsive in the office over a robust network connection may slow to an unusable crawl over a poor connection.

You only really understand the potential issues – and solutions – when you get in the field with mobile workers, experience all aspects of their environment, and observe the sequence of their workflows.

Teams constantly on their feet will want light-weight devices with long battery life. People collecting data in sporadic bursts, like inspectors, won't want to wait for devices to boot up. For quick, ad-hoc data capture, they'll need fast-starting devices or to continue using paper, which they can digitize instantly with digital pens or back in the office with scanners.

We've run into a wide range of unexpected environmental issues that resulted in expensive solutions failing in the field. Workers wearing gloves, for example, typically don't want to use solutions that require them to constantly take off their gloves to enter data. In another example, a pipeline construction firm abandoned a tablet solution, when it became clear that the only way field teams could view screen in the bright Gulf sunshine was by shielding it in unassembled pipe sections.

The good news is that there are more device options than ever for teams to choose from.

To avoid this gotcha, include mobile workers in your initial tests or pilot. It's also important to ensure that the team involved in the pilot has an interest in the project's success.

You may get skewed results from doing a pilot or proof of concept with a team that is uninterested in the outcome or overwhelmed by other tasks. Uninterested testers may also be less forthcoming with feedback or less forgiving with frequent tweaks to the approach as you fine-tune the experience during testing.

Early user-acceptance testing with the right team of engaged workers may also result in inspiration or ideas for ways to automate and streamline different business processes. If your team is considering radically different device options, then you can test multiple solutions for the same workflow using two different teams. Simultaneous testing of different variations may accelerate results or reveal meaningful differences in how to best meet the needs of different workers or environments.

Gotcha #2:
Getting locked into a proprietary solution that stakeholders – such as IT – don't like

Along with the many device options – there are also many software options for building mobile apps.

One of the key avoidable gotchas is selecting a proprietary solution that few people know how to implement and support. I've run across many teams that implemented proprietary solutions that required unexpected training burdens in order to build and update apps.

Many have also faced IT resistance – since resource-strapped IT teams can be reluctant to support unfamiliar apps or bring them on their networks. It's especially challenging when it comes time to fix or change apps. Very often, teams find that the only way to implement fixes or changes to proprietary apps is to rely on employees who may have changed roles or go back to the vendor for more expensive consulting.

The key to avoiding this gotcha ▸ collaborate with IT to pick an approach that is built on standard tools and data formats.

With the emergence of the Cloud and employees bringing their own devices (BYOD) to work, many operation teams have been empowered to make more and more of their own technology decisions independent of IT. Even though many apps can now be developed and deployed on devices or through the Cloud independent of IT teams, it still makes sense to include IT early in the process. Regardless of how you implement your solution – your IT team may ultimately want or need to get involved in your mobile project to extract data or even to support devices. To make that process smooth – get their feedback early.

IT's early involvement can help avoid such gotchas as teams selecting browser-based app development that is incompatible with their future planned device purchase. IT can also help surface issues with vendors and data architecture approaches that can make it difficult to extract or integrate data into key internal back-end systems and processes in the future.

Proprietary solutions lead to proprietary issues. The likelihood of hitting these types of issues gets reduced dramatically as teams work with vendors and solutions that are based on familiar, standard tools, generating standard file formats and methods for accessing data.

Gotcha #3:
Not preparing for change

Even in the fast-moving mobile app world – it's surprising how many developers build and deliver hard-coded apps and expect that their work is done.

With even the best app, the more it's used, the more people think about improvements and send change requests. While change requests can be a pain, active change requests can also be a good reflection of mobile workers using, engaging, and wanting to improve the app. It's even better when they start requesting new apps, forms, and mobile workflows to automate.

The gotcha comes when you realize that your hard-coded solution can't be easily updated or tweaked. You've painted yourself into a corner. Many teams are shocked when they return to vendors or custom app developers to make change requests. Even minor changes often result in major expenses.

The key to avoiding this gotcha ▸ work with vendors and solutions that can be changed and updated relatively easily.

Build change into your original plan – from the perspectives of technology, vendor, and budget. Complexity often comes in waves. Solutions that are complex to build, also tend to be complex for training, to use, and to update. When there's lots of initial custom coding and scripting, then making changes later can often seem as if you are rebuilding (and paying again for) the entire app.

In an ideal world, operational groups could easily make changes themselves – without creating a burden on IT or requiring expensive change fees. That ideal world starts with familiar form design tools. Ideally, anyone who can design a form in Excel should be able lay out a mobile form, enabling them to easily create and change forms.

A good test of a vendor and their form design tools is to send them one of your forms and ask them to do a quick live demo. If it takes a lot of time or requires too much money up-front for a pre-sales demo – then that may be a future glimpse into the complexity of using their tools.

Of course making the changes is only part of the issue. For many teams, it's also difficult to get changes pushed out to the mobile workers in the field. Any solution that requires devices to be physically gathered and handled by IT can be very difficult to manage – especially for larger organizations with distributed teams.

The easiest apps to manage are ones that are entirely web-based, since changes can be made and managed on the server. If apps need to be usable while devices are offline, then offline client apps that can be managed and updated remotely are the best choice. It's getting easier with new web-based technologies, which can support rich apps usable online or offline – all through the browser.

Of course the best approach will depend on a team's specific needs. The key is for teams to plan for change and make sure that their solution can support the likely change requests and needs that they will encounter.

FINAL THOUGHTS

There is no one-size-fits-all in mobility. Successful deployments require teams to consider a variety of factors. While each technological approach has its own unique set of specific considerations, the three gotchas above can rear their heads in any mobile technology deployment. The good news is that they can be easily avoided.

I hope that you've found this eBook useful. While every mobile deployment is different, understanding these gotchas and opposing best practices will help you find the right solution for you and your team.

Please reach out if you have any comments, questions, or would like to get some feedback on your workflow. I currently work at Field Data Integrators™ with a team of workflow specialists that are all about finding the right fit. We've fielded thousands of deployment questions and requests, some are perfect fits for the digital pen and tablet workflows that we support, others we have directed toward other solutions based on their workflow needs.

Have a question? **Click here**
sales@fielddataintegrators.com
to reach out and email us.