Top 5 ways you can improve your iOS app's Bluetooth communication

The nature of Bluetooth means that some of these topics will get a bit technical (especially #5!). I've tried to explain them in as much "plain English" as possible, but you might find it helpful to pass these along to your development team for deeper review.

#1 - Good Error Handling

Wearables are a very intimate experience for your users. You are asking them to wear a piece of technology on (or even in) their body. If that wearable isn't flawless (it won't be) and the user is left feeling uninformed and frustrated they will be quick to toss it in the top dresser drawer.

Too many apps throw a generic "An error occurred, please try again." message when anything Bluetooth or sync related happens. Provide a specific action so the user feels like they have some control over the situation. Something like "Sorry, we were unable to complete a sync of your data. Please make sure your band is close to your phone and try again." is better than the first error message.

#2 - Logging

There's no denying it, Bluetooth can be very difficult to debug. When working with radio signals like Bluetooth the environment your wearable device is in can have a big impact on the performance and reliability.

Just because you have a great experience "on the bench" doesn't mean your users will have the same experience. This can leave your team frustrated and lost. Recreating user issues can be difficult or impossible, making finding and fixing those issues feel like a giant time-suck.

For these reasons and many others, being able to find and debug issues from the field is crucial! The best way to gain insight into how your wearable behaves in the field is through copious logging. Meaning your app should be writing a lot of information to a file that you can retrieve from a user's phone if there is an issue that your team needs to look at.

Usually, the first lines of code I change for clients are around logging. It's that important! I make sure every step is well recorded in the logs. Recording everything that happens from connection until the data is at rest (written to a file, DB, etc). If anything happens during communication with a wearable I want to know exactly where it went wrong.

#3 - Automatic Retry

Given the unpredictability of wireless signals, there are often transient conditions that can cause temporary errors. This is often seen as a failure in the Bluetooth connection to the wearable, but can also be a failure of an individual transaction. Rather than immediately failing and notifying the user, it is better to first retry.

It requires a building a bit more robust Bluetooth communication stack, but it will provide a much smoother experience for your users. For the systems that I build, I will usually allow for 3 retry attempts of either a transmission or the connection before throwing an error to the user.

#4 - Advertising Interval

The advertising interval is the frequency with which a Bluetooth peripheral (generally this is the wearable) will advertise itself when it is not already connected. The iOS (or Android) device will scan for peripherals nearby on a specific interval, and will only scan for a certain period of time. These two things must overlap in order for your app to "see" your wearable.

The advertising interval is often compromised to save on battery life of the wearable. However, that can have the negative side-effect of causing a lot of failed, or very long, connection attempts from your app to your wearable.

If your app is able to consistently find and connect to your wearable this may not be an issue worth pursuing immediately. However, if it is a problem you're experiencing then the advertising interval of your wearable would be the first place I would look.

Apple has published very specific recommendations on the best advertising intervals to work with the scanning behavior of their devices. Their guidelines can be found at: https://developer.apple.com/hardwaredrivers/BluetoothDesignGuidelines.pdf

#5 - Services and Characteristics UUIDs

A GATT profile containing services and characteristics is how Bluetooth defines communication between your wearable and your app. These services and characteristics all use alphanumeric addresses (a UUID) to identify themselves. The Bluetooth SIG (Special Interest Group) has created a pre-defined list of profiles for common needs (heart rate, temperature, running speed and cadence, etc).

If you wearable has data that fits one of the pre-defined profiles, I would recommend utilizing them unless there is a business reason not to.

Otherwise, you will be creating a custom profile with your own addresses. Way too often I see wearables implementing the short version (16bit) of addresses. While technically OK if the address is in an “open” range, it can cause some odd issues when you bump into other wearables with the same address. It’s not unlike two houses having the same address and getting their mail mixed up.

The proper way to create addresses for your profile is to use the longer (128bit) addresses. The chances of a collision with another wearable is near 0, saving yourself from chasing odd connection issues that only happen randomly.

BONUS #6 - Smooth and Helpful Onboarding

Successfully onboarding users is an often overlooked, but very powerful, part of setting your users up for success and higher retention! First impressions are everything, and I see too many apps that do this poorly.

The hurdle for onboarding to a wearable app is even higher than many other apps. For one, there is often a lot more personal information that must be collected from the user. For another, users are really onboarding two different things that need to communicate; both the wearable and the app.

The onboarding process is your chance to begin educating your user on how your wearable functions, and what they can expect in the days and weeks to come. Use onboarding as your chance to slowly introduce users to the various functions of your wearable. These are things like buttons, touch screen, progress and achievement notifications, etc.

NEXT STEPS

Thanks for taking the time to read through this guide. I'm flattered!

#1 - Bluetooth Assessment

Would you like me to personally take a look at your app and wearable communication, and make recommendations specific to your situation? I offer a Bluetooth Assessment just for that.

http://five3apps.com/bluetooth_assessment

#2 - Work directly with me

If you would like to work directly with me for any of your wearable iOS app needs — whether it's Bluetooth, onboarding, user retention, or building your MVP — you only need to ask. You can find out how to work with me by visiting my website at: http://five3apps.com