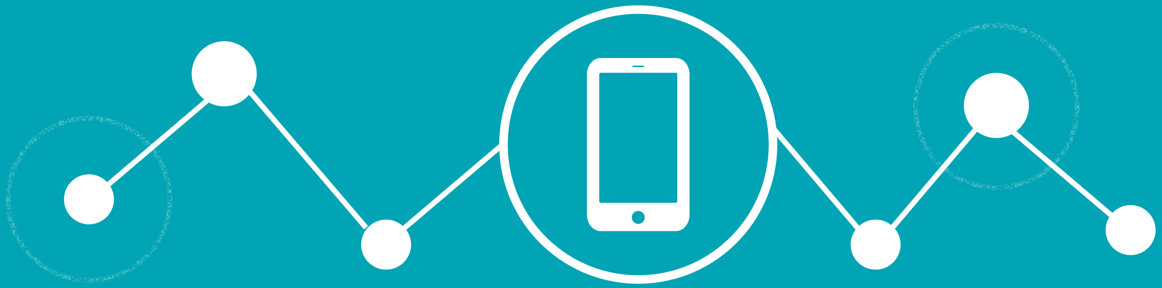
Designing Connectivity INTO MEDICAL DEVICES



SUNRISE LABS

SUNRISE LABS

Table of Contents

The Allure and Reality

Adding wireless connectivity to devices is a trend rapidly gaining momentum in the healthcare industry. Benefits can include reduced costs and improved outcomes for patients and healthcare providers.

Smartphones and tablets offer nearly everything needed for a modern and wireless user interface—high-resolution touchscreen, Wi-Fi/Bluetooth/cellular connections and a powerful, low-energy processor—all in one compact, reasonably priced package; right?

The concept of a low-cost, plug-and-play interface sounds ideal—but it's critical to remember that smartphones and tablets are consumer, not medical devices. Integrating a commercial phone or tablet into your medical device involves much more than wiring a few connections and loading software.

This paper will help you evaluate your options and address some less frequently identified development issues when considering ways to integrate wireless connectivity within your product.

Integrating connectivity components changes your solution from a device to a system. Broader systems considerations such as accommodating Bluetooth /wireless technology, sensors, cloud connectivity, and database architecture, along with implementation of cybersecurity protocols and battery and power management techniques all need to be considered. Tradeoffs often are involved to create the solution that meets program goals. For example; modifications to Bluetooth range or sensors could impact the device's overall battery life.

As devices become more integrated with healthcare systems, cybersecurity challenges are no longer confined to the device itself. This necessitates device manufacturers and designers to address security challenges to effectively mitigate cybersecurity vulnerabilities. In terms of smartphones and tablets, it's critical to keep user security, safety, and compliance in mind when choosing a platform (i.e.; Android, iOS, or Windows) as there are numerous pros and cons to each.

First Things First

The most important step is to define the requirements of your device as completely as possible at the beginning of the development process. Everyone on the product team should thoroughly understand the goal before development begins.

Don’t make the mistake of taking your eye off of FDA regulations and the approval process throughout the entire design and development effort. This will ensure compliance and avoid last-minute surprises that could delay time to market. If you lack in-house expertise, a design firm that thoroughly understands and has extensive experience with the FDA process will be a tremendous help.

Once you’ve defined your goal, you’ll need to address several other critical areas.

Design Life

Most medical devices are in production for several years, yet the design life of a phone or tablet is typically less than 12 months. With medical devices under formal design and manufacturing controls, changing phone or tablet models will require significant verification and process validation testing—on top of the engineering needed to make the device work. You'll also need to look at maintenance and field support complications from having multiple models in use. If you have the capital, a large one-time buy is a way to limit these issues. But, be careful—phones and tablets are expensive components, and their value will drop quickly if they are not commercialized soon.

Software Distribution and Control

How is the phone or tablet's software distributed? iTunes or Google Play? Downloaded over the Internet? Or via hardwire, USB, WIFI or Bluetooth? Distribution of medical device software needs to be carefully considered. There are challenges with each of the methods listed above. iTunes requires the application to comply with Apple's style guide. Google Play, the Internet, a hardwired connection, USB, WIFI and Bluetooth may all require security measures to ensure the software is properly installed. In addition, compliance issues may arise to ensure the software installation is recorded with the manufacturer.

Safety, Security and Compliance

Safety and security features must be reviewed from the perspective of industrial software and design. Security features are a given, but users may also encounter viruses, spyware, Trojans and denial of service. Compliance is another critical area, and all applicable standards must be listed as part of the requirements. With hardware standards like IEC-60601, standards for security, standards for user interface development, standards for Europe and standards for the US, what you don't know or fail to address early will lead to costly surprises later on.

Licensing

Is open source software used in the smartphone or tablet? Any applicable licenses will need to be displayed on the user interface.

Packaging

Will the off-the-shelf phone or tablet need additional protection to withstand being repeatedly dropped or cleaned in clinical settings? What IFUs (Instructions for Use) are available, and can they be accessed through a user-friendly application?

So Far, So Good

Has the phone or tablet you're considering accommodated all of the constraints discussed above? If so, it may work for your application. Here are some specifics to consider before you decide.

Choosing a Platform

There are three primary platform choices: Apple's iOS, Google's Android and Microsoft's Windows-based platforms. Each has its own advantages and drawbacks.

Platform	Key Advantages	Key Disadvantages
Apple iOS	• Ubiquitous • Good security • Fewer platforms to test • Tightly controlled by Apple	• Should comply with Apple's style guide • Developers are more expensive
Android	• Ubiquitous • Lots of drivers and available software • Google is backing the software	• Many platforms to test • Security is difficult • Open system • Free software may be poorly written • May need operating system knowledge to build system
Windows-based	• Well known interfaces match the PC • Developers are easy to hire	• Known security vulnerabilities • Security being challenged regularly • Software quality may not be acceptable for a medical device

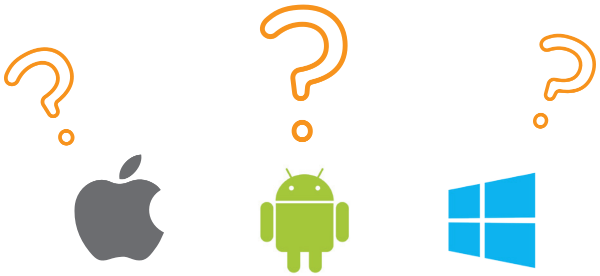
Apple iOS. The Apple operating system is familiar to a large audience. Development rules are well-defined. Apple maintains tight control over their products, so there are fewer platforms to test. The got-cha? Software distributed through Apple must adhere to their look and feel, and Apple has final say on what's distributed through their store.

Android. Unlike Apple's tightly controlled environment, Android is an open system. Android devices are generally less expensive than Apple products, and available in a wider number of configurations. The watch-out? More platforms to test.

Windows. Many users have a love/hate relationship with Windows 8. Windows 8 is based on Microsoft's well-known OS. Like Apple's iOS, Windows 8 can be coordinated across computers, laptops, phones and tablets. The drawback? The hardware selection is currently smaller and based on Microsoft's PC platform, which is frequently targeted by viruses.

Is there anything else? The answer is yes! A fourth option is to work with a carefully vetted design team to develop your own platform. This is more feasible than it sounds at first. An experienced contract R&D firm will allow you to incorporate and customize features you want, take out ones you don't, and hardwire critical functions for greater reliability.

With any platform, pay careful attention to hardware performance. What's the battery life and temperature range? Is screen resolution adequate for the intended use? What IPX rating must it pass? Does it have a CE mark? How long will this model be available? What are the emissions and susceptibility?



Choosing an Operating System

The next step is to determine which OS version to install. Start by considering all the implications.

Configuration control. The OS image may need to be kept under configuration control so that each time the software is manufactured, the same version of the OS is distributed. If not, the downside is a re-test of the phone or tablet to ensure all functions still work as previously tested.

Licensing and trademarks. Should licensing and trademark issues emerge, you'll need knowledgeable (and expensive) legal counsel. Remember that any changes made to the Android OS must be submitted back to the Android community, and some of the application software may need to be distributed.

Configurability. The OS may need to be configurable to deal with issues that commonly arise during development, such as internationalization, disabling features for security and vulnerability, using cloud services and handling HIPAA regulations. Be sure the OS contains a variety of fonts that provide a good selection for internationalization, licensing and UI design

Other Considerations

Internationalization. Most medical devices require their software to go through an internationalization process so that it can translate text, change screen symbols and modify colors for market acceptance. Text layout can be a big problem—for example, German text does not usually fit, and some countries require text to be written right to left, with numbers displayed left to right. Development must address the display of dates, punctuation, time format and date/time boundaries. Much of the software for the platforms mentioned above support internationalization. Failing to design these factors in at the beginning may require extensive re-writing in subsequent releases.

Manufacturing concerns. Don't be blindsided by manufacturing-related problems. If you choose an Apple, Android or Windows device, its manufacturer must be capable of loading the correct OS image and application. If necessary, a signed application can be loaded and the device locked, so that any subsequent software changes will render the device non-operational.

With third party devices, hardware changes may affect the software by making capabilities that were previously not operational unexpectedly become active.

Almost there...

Developing software for medical devices is a difficult task. Understanding the additional design aspects needed to support integration with a commercial phone or tablet requires highly specialized experience.

How will you handle software safety and risk? Creating software with a safety class that detects corruption may be critical to preserve a variable and identify when corruption occurs. Display of the value typed in by the user may require a design where the value is "round-tripped" to the controlling device before being re-displayed for confirmation.

Is the software intended to be fail-safe or fail-operative? For Class II and Class III medical devices, risk is a crucial component of the development process, and software architecture must be designed with attention to risk-management. How will the device behave if a failure occurs during communication with the embedded device, or within the phone or tablet itself?

Have you considered software testing? Testing software for use in medical devices involves additional cost and expertise that may not be available from in-house software development teams.

As part of the development process, how much and to what degree should integration testing be performed? What about system software testing, as well as security and vulnerability testing and design?

Like all computer technology, phones and tablets are vulnerable to cyber-attack. Current FDA guidance on the subject is minimal. Certain features, such as Wi-Fi, cellular, BLE, Bluetooth and infrared, may need to be disabled to prevent intentional harm from hackers. The mechanism for disabling these features must be thoroughly documented. This is part of the risk assessment, and the FDA will want to know how device security and vulnerability are handled.

Other vulnerability issues include theft, side-loading of the OS or unwanted images, viruses, Trojan horses, and denial of service. The software may or may not be able to run if attacked, but risks must be documented and software should not cause any harm.

Have you thought through the software submission process? A concise, well-organized submission for a 510(k) or PMA will depend on having a well-defined goal and development process. Without proper risk identification and comprehensive testing, your submission will be incomplete and the FDA may require more information—which can delay and possibly even de-rail your product launch.

The Finish Line

Smartphones and tablets have great potential as medical device interfaces, but they are not simple drop-in solutions. Options range from semi-custom to fully custom platforms. Each has pros and cons relative to cost, engineering, time to market and feature set. It is critical to fully understand the trade-offs around architecture, technology, operating systems and software.

Sunrise Labs would welcome the opportunity to use our inside knowledge to help you identify the platform and software that best meet your technical and business needs, and help you successfully navigate the FDA approval process.

About Sunrise Labs

For over 25 years, clients have come to Sunrise Labs for complete product development, project rescue and engineering services, leveraging our ISO-13485 certified process. Client success stories reflect our strengths in system engineering, project management and a full range of technical disciplines to turn novel ideas into commercially viable products. We are known for solving tough engineering problems nimbly and with integrity. Our team has deep experience in medical device development, ranging from monitoring solutions that include portable, wireless, battery powered devices connected to mobile devices and back end systems to complex instruments with high-performance analog conditioning, closed loop controls, real-time, multi-tasking software and intuitive user interfaces. Please contact us today to learn more!

www.sunriselabs.com

SUNRISE LABS

5 Dartmouth Drive
Auburn, NH 03032
(603) 644-4500
www.sunriselabs.com