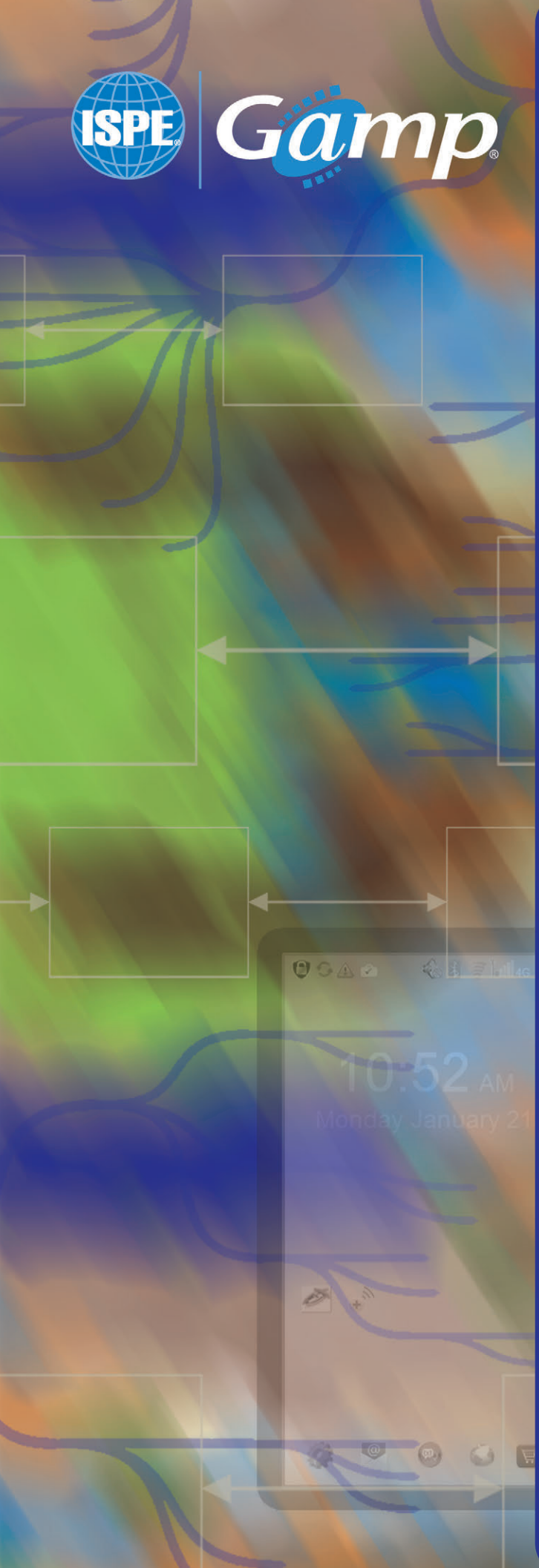
ISPE Gamp

GAMP Good Practice Guide

A Risk-Based Approach to Regulated Mobile Applications



GAMP Good Practice Guide

A Risk-Based Approach to Regulated Mobile Applications

Disclaimer:
This Guide is intended to provide a risk-based approach to implementing and supporting regulated mobile apps. ISPE cannot ensure and does not warrant that a system managed in accordance with this Guide will be acceptable to regulatory authorities. Further, this Guide does not replace the need for hiring professional engineers or technicians.

Limitation of Liability
In no event shall ISPE or any of its affiliates, or the officers, directors, employees, members, or agents of each of them, or the authors, be liable for any damages of any kind, including without limitation any special, incidental, indirect, or consequential damages, whether or not advised of the possibility of such damages, and on any theory of liability whatsoever, arising out of or in connection with the use of this information.

All trademarks used are acknowledged.

ISBN 978-1-936379-75-0

Preface

The use of mobile platforms – including smartphones, tablets, and wearable devices – by the general public has increased hugely. In 2013, sales of smartphones exceeded the sales of standard mobile phones, and this trend shows no sign of slowing. As reported on the FDA website, according to industry estimates, 500 million smartphone users worldwide will be using a health care application by 2015, and by 2018, 50 percent of the more than 3.4 billion smartphone and tablet users will have downloaded mobile health applications. These users include health care professionals, consumers, and patients. With this much computing power in everyone's pocket, it was inevitable that regulated companies would recognize an opportunity to use it for a variety of purposes. Some examples are:

- To improve patient compliance to a medical regimen
- As a marketing tool
- To make medical literature more available to physicians
- As a channel for patient reporting
- As an interface to a medical device
- To control a medical device

In addition to the above, mobile devices are an attractive option as interfaces to GxP systems in a manufacturing plant or laboratory, e.g.:

- Tool for warehouse management, including goods receipt or movement
- Dashboard interface to a large number of laboratory systems
- Interface to manufacturing equipment, possibly with the ability to adjust setpoints
- To access enterprise applications, such as Enterprise Resource Planning (ERP)

The universality of the mobile device market makes it increasingly attractive to develop mobile applications for varied uses. It may not be immediately obvious to all teams and departments that there are regulatory implications. Hence building awareness within all affected areas of regulated companies is critical.

By their very nature mobile devices present a significant challenge to control. The possibility of putting regulated applications, some of which may be classified as medical devices, into the pockets of the public is new ground for the industry. Never before has regulated software run on platforms where life science companies have little or no control over the platform.

Acknowledgements

The Guide was produced by a Task Team led by Arthur (Randy) Perez (Novartis Pharmaceuticals Corporation). The work was supported by the ISPE GAMP Community of Practice (COP).

Core Team / Chapter Leaders

The following individuals took lead roles in the preparation of this document and each managed one or more chapter teams made up of writers and contributors.

Erik Anderson	Johnson & Johnson Family of Companies	USA
Niklas Bergvall	Oxford Instruments	United Kingdom
Winnie Cappucci	Bayer AG (retired)	USA
Christophe DuPont	Baxter	Belgium
Arthur (Randy) Perez	Novartis Pharmaceuticals Corporation	USA
Sion Wyn	Conformity Limited	United Kingdom

GAMP Editorial Board

The Core Team wish to thank the following individuals for their valuable contribution during the preparation of this document.

Chris Clark	Bard Pharmaceuticals Limited	United Kingdom
Colin Jones	Conformity Limited	United Kingdom

Regulatory Input and Review

Krishna Ghosh	U.S. Food and Drug Administration	USA
Stephen Grayson	Medicines and Healthcare products Regulatory Agency (MHRA)	United Kingdom
John Murray	U.S. Food and Drug Administration	USA
Phillip Pontikos	U.S. Food and Drug Administration	USA
Robert Tollefsen	U.S. Food and Drug Administration	USA

Subject Matter Expert Input and Review

Particular thanks for substantial and valuable contributions to the review process go to:

Chris Price	Aberystwyth University	United Kingdom
David Stokes	Venostic Solutions Group Limited	United Kingdom

The Team Leads would like to express their grateful thanks to the many individuals and companies from around the world who reviewed and provided comments during the preparation of this Guide; although they are too numerous to list here, their input is greatly appreciated.

Company affiliations are as of the final draft of the Guide.

Connecting a World of
Pharmaceutical Knowledge

ISPE Headquarters
600 N. Westshore Blvd., Suite 900, Tampa, Florida 33609 USA
Tel: +1-813-960-2105, Fax: +1-813-264-2816

www.ISPE.org

Table of Contents

1 Introduction

1.1 Background

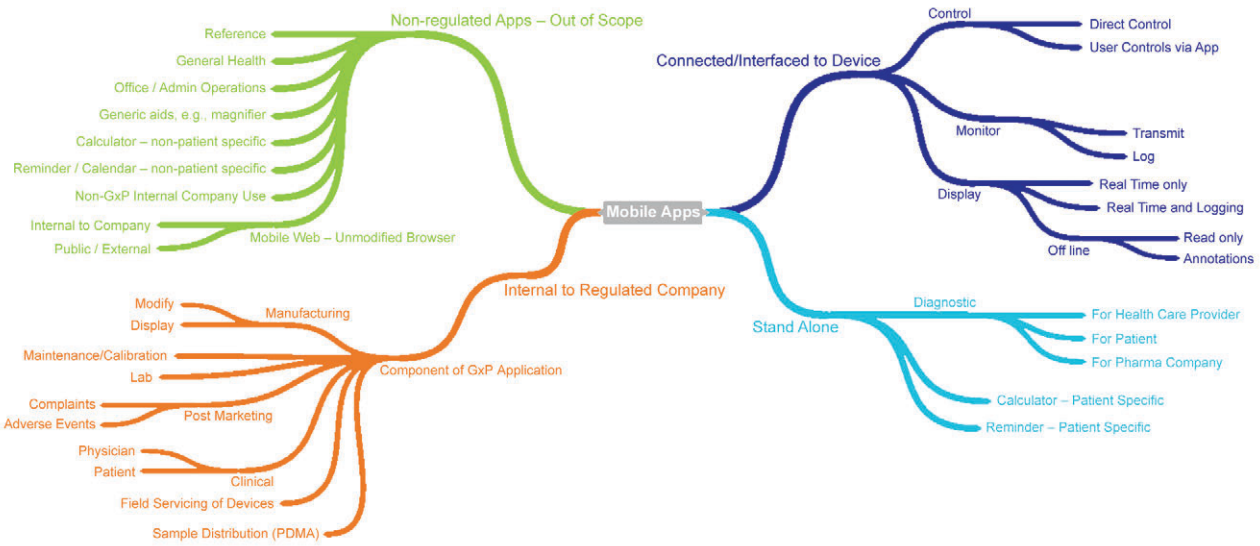
The development of mobile computing, particularly the increasing use of smartphones and tablets, has led to a significant change in how computers are used. A large proportion of the population can now carry considerable computing power in portable personal devices, and it was inevitable that these would be increasingly used within the life sciences industry.

In some cases, this will lead to regulatory implications as mobile applications may be subject to GxP regulations[1] (mobile apps).

For example, it is clear that the iPhone-based electrocardiograph approved by the U.S. Food and Drug Administration (FDA) in 2012 is a medical device, and that it requires a wide variety of controls just like other sophisticated electronic medical devices. However, many mobile applications are created for other purposes, such as support for marketing. Regulators have recognized an increased use of mobile medical apps, as illustrated by FDA's publication of guidance on the topic in 2013 [3]and the MHRA's guidance issued in 2014 [4].

Figure 1.1 provides an indication of the wide variety of different types of mobile apps that may be used within the life sciences industry.

Figure 1.1: Example Types of Mobile Apps



In general, unless a mobile app is used only for viewing static information, e.g., medical literature, it is likely that the mobile app would be regarded as regulated. Regulators have recognized this, and have begun inspecting and approving a variety of mobile apps as medical devices. Mobile apps also may be used as interfaces to GxP regulated instruments and equipment, and as components of GxP regulated computerized systems.

[1] GxP regulations are the underlying international pharmaceutical, medical device, or other life science requirements, such as those set forth in the US FD&C Act [1], US PHS Act [2], FDA regulations, EU Directives, Japanese regulations, or other applicable national legislation or regulations under which a regulated company operate. See Glossary for full definition.

The largely uncontrolled nature of the mobile platforms, upon which such regulated mobile apps may run, present a challenge to companies wishing to use them. For traditional validated applications, one element of maintaining a state of control is controlling the platforms upon which they run (e.g., through formal qualification). When the platforms are mobile phones used informally by members of the public, the control of the platform becomes a significant challenge and qualification is not possible.

Regulatory input for this document was taken from five main sources:

1. Guidelines on the Qualification and Classification of Stand Alone Software Used in Healthcare within the Regulatory Framework of Medical Devices, European Commission DG Health and Consumer, January 2012 [5]

2. Regulation of Medical Software and Mobile Medical "Apps," Australian Therapeutic Goods Administration, 13 September 2013 [6]

3. Guidance on Medical Device Stand-Alone Software (including Apps), UK Medicines and Healthcare Products Regulatory Agency, 19 March 2014 [4]

4. Medical Information Systems – Guidance for Qualification and Classification of Standalone Software With a Medical Purpose; Swedish Medical Products Agency, 31 January 2013 [7]

5. Mobile Medical Applications: Guidance for Industry and Food and Drug Administration Staff, US Food and Drug Administration, 25 September 2013 [3]

1.2 Purpose

This Guide is intended to provide a risk-based approach to implementing and supporting regulated mobile apps. It addresses the following issues:

1. Deciding whether or not the mobile app is regulated

2. Understanding when a regulated mobile app may be a medical device

3. The unique risks related to software that is partly managed by users

4. The implications of constant connectivity (or the loss thereof)

5. The complicated issues of managing software on a variety of operating systems at different version levels on many different devices

6. Ensuring compliance with applicable regulations, including implications for data integrity and data privacy/ protection

7. Recommendations for retiring mobile apps

This Guide applies the principles and practices described in GAMP® 5 [8] to the novel and unique challenges of the mobile environment, to provide the users with safe and effective solutions.

Note: for the purpose of this Guide, the term **users** include health care professionals, patients, and individual members of the public.

1.3 Scope

This Guide covers two types of mobile apps:

1. Mobile medical apps, i.e., mobile apps that meet the definition of a medical device and are intended to be used in one of two ways:

    i. As an accessory to a regulated medical device (e.g., for remote display of data from bedside monitors)

    ii. To transform a mobile platform into a regulated medical device

(These correspond to mobile apps either connected or interfaced to a medical device, and stand-alone apps, as shown in Figure 1.1. These mobile apps require a complete life cycle approach as described in this guide).

2. Mobile apps that are used as part of GxP operations at a regulated company, as a component of a GxP regulated computerized system, such as an interface to an instrument or control system

(These mobile apps will typically be addressed as part of wider system validation or qualification activities as defined in the regulated company Quality Management System (QMS), taking into account the novel and particular aspects of mobile apps as described in this Guide).

This Guide covers a variety of application types that run on mobile platforms, including native apps and web apps for mobile devices.

The term *mobile platform* typically refers to smartphones and tablet computers. The majority of such systems will run on one of three operating systems:

1. Apple iOS®

2. Google Android®

3. Microsoft Windows®

Mobile platforms using other operating systems are also in scope. While all smartphones are capable of communication via cellular radio, tablet devices may or may not have that capability. However, most mobile devices do have Wi-Fi™ connectivity.

This Guide does not focus on the mobile platforms themselves although it does acknowledge and address their importance. Unlike traditional infrastructure, it may be difficult for a regulated company to exert control over the hardware or operating system. Mobile platforms are commodity items and are typically the property of individuals, rather than being corporate assets. This Guide focuses on the mobile app software development life cycle and software or procedural controls.

Some of the challenges presented by the inability to control the configuration of a user's mobile device need to be met with innovative controls, e.g., a "handshake" approach where specific software checks are executed prior to opening the mobile app. The need for such compensating controls should always be risk-based.

This Guide does not cover medical devices to which the mobile app may be interfaced, nor the software (e.g., firmware) included in such medical devices.

The Guide is intended to show how current industry good practice as described in GAMP® 5 [8] may be applied to mobile apps. The Guide is not intended to cover detailed requirements for medical device classification, registration, development, or support.

Detailed requirements for medical devices will vary from region to region, and it is the responsibility of organizations and individuals involved to identify, interpret, and apply such requirements as applicable. Companies should be familiar with local regulations in any region where they intend to deploy a regulated mobile app. Relevant international standards, such as ISO/IEC 62304 [9] and ISO 13485 [10], also should be consulted.

1.4 How to Use this Guide

This Guide contains this introduction, a main body, and a set of appendices. It has been structured to meet the needs of various readers, and contains, in increasing level of detail:

1. An overview of aspects and risks specific to mobile apps, along with a high level overview of how to address them

2. Further information on how to apply the GAMP® 5 [8] life cycle and Quality Risk Management (QRM) approach to mobile apps

3. More detailed “how to” guidance for specific topics

4. Example case studies

Readers requiring an overview of the topic should read this Introduction, plus Section 2, Specific Aspects of Mobile Apps, and Section 3, Application of GAMP® 5 [8] to Mobile Apps.

Readers seeking further information on the overall mobile apps life cycle, also should read the remaining sections of the main body.

Readers requiring detailed information on developing and supporting mobile apps also should consider the applicable appendices, based on their areas of interest and responsibility.

Suppliers of mobile apps should find Section 2 Specific Aspects of Mobile Apps, Section 3 Application of GAMP® 5 [8] to Mobile Apps, and Appendix 6 Supplier Management and Good Practice, particularly relevant.

All readers may find the example case studies in Appendix 7 useful when applying the guidance provided to specific situations. The case studies reflect a wide variety of types of mobile apps.

2 Specific Aspects of Mobile Apps

Mobile platforms and mobile apps provide a rich and easily accessible toolset of functionality to users, which raise a number of questions in a regulated environment, such as those noted in Figure 2.1.

Figure 2.1: Example User Interface of a Mobile Device

This section addresses the novel and specific aspects of mobile apps that require specific consideration and action. This section also refers to other sections within the Guide that provide further guidance on these topics.

Tables 2.1 and 2.2 summarize those novel and particular aspects. Table 2.1 highlights topics related to management of mobile app development and operation, and Table 2.2 covers technical topics related to functionality and design.

Table 2.1: Topics Related to Management of Mobile App Development and Operation

Area	Aspect	Risk Factor	For More Information
Project Governance	Mobile App Classification	**The need to identify whether the proposed mobile app is regulated.** The project team may not have the knowledge, experience, or understanding of whether the mobile app is GxP regulated, and whether it is classified as a medical device.	Section 5.2
	Supplier Related	**Some mobile apps will be developed by very small development companies, e.g., "garage developers."** While this in itself is not a risk, the chance is greater that very small development companies will not understand the expectations of regulators and client companies.	Sections 6.2, 6.3 (Appendix 1) Appendix 6
	Supplier Related	**Software developers from small companies may be less financially stable.** If a mobile app is intended to be used for several years, special measures may be necessary to preserve software support services should the developer company fail.	Section 6.3 (Appendix 1) Appendix 6
	Project Goals	**Not all regulated mobile app projects start out with the intent or expectation of building a regulated app.** Mobile apps may be generated from a marketing idea for a patient or physician aid. During the project, it is recognized that the product will be a mobile medical app. If the initial phase of the project has been managed at a low level of formality, documentation that will meet medical device standards may be a particular challenge.	Section 5.2
	Project Team	**The mobile app project team may not have appropriate Subject Matter Experts (SMEs).** Where a mobile medical app was not the original intent, the project team may not have the appropriate expertise to produce a device that meets appropriate quality standards and regulatory requirements. In this case, SMEs should be added as soon as the regulatory relevance is recognized, and the team may have to remediate some of their initial output.	Section 5.2 Section 6.2 (Appendix 1)
	Project Size	**Mobile app projects may be very small.** Depending on the nature of the mobile app, development time may be as little as two to three weeks. As a result, mobile app project teams tend to be small, but with very aggressive timelines for completion. This can be a challenge for creating and approving documentation. Where the project is not recognized as GxP relevant from the point of conception, the short timeline can magnify the issue.	Section 6.3 (Appendix 1)
Operational Governance	Change Control	**In general, users have limited ability to address a problematic change.** If an upgrade fails, users may have no mechanism to back out a failed change.	Section 6.4 (Appendix 1)
	Calibration	**Some mobile apps may perform calculations based on data from attached input devices.** This functionality may require initial and periodic calibration to verify accuracy.	Section 5.2 Appendix 3
	Retirement	**It may be difficult to withdraw a mobile app from the market.** The mobile device may not belong to the mobile app provider. Users may continue to use retired versions of mobile apps until they replace their current device. In addition, there may be data residing on the device that should be retained, but does not result from user actions.	Section 5.2 Section 6.5 (Appendix 1)

Table 2.2: Technical Topics Related to Functionality and Design

Area	Aspect	Risk Factor	For More Information
Design	Configuration	**Configuration requirements may vary between platforms.** For example, the ability to enable/disable audible alarms may be required, but may not be available on all platforms. In addition, the possibility of multiple configurations across platforms may introduce risk.	Appendix 3
	Browser	**There may be specific browser requirements to consider.** This could affect the use of the mobile app.	Appendix 3
	Messaging	**The mobile app may be required to send and receive e-mails and text messages.** This may impact the performance, security, and availability of the mobile app.	Appendix 3
	Clock / Calendar	**The mobile app may depend on accurate date and time functionality.** Critical records may need to be time and date stamped. Consideration should be given to the source and control of this functionality.	Appendix 3
Data	Data Management	**Data may reside on a device outside the regulated company's direct management.** Such devices probably will not be subject to the same level of control.	Section 5.3 Section 6.4 (Appendix 1)
	Data Storage	**Data on mobile devices is at risk.** Mobile apps may store records on the mobile device (long or short term). While on, the device records are more vulnerable to accidental or malicious loss or alteration. If timely transmission or synchronization of medical information is important and under control of a user, this also introduces risk.	Section 5.3 Section 6.4 (Appendix 1)
	Data Integrity	**Data integrity may be compromised by user action or inaction.** If a device is lost, damaged, or otherwise compromised, it is possible that important records residing on the device will be irretrievably lost. Access control may be problematic. For example, users trying to improve performance of their devices may destroy records by either deleting a mobile app or clearing memory. There are no restrictions on administrative rights.	Section 5.3 Section 6.4 (Appendix 1)
	Data Privacy	**The mobile app may originate, maintain, and/or process private data.** All applicable legislation may not have been considered in order to ensure the appropriate management of private and sensitive data.	Section 5.3 Section 6.4 (Appendix 1)
Usability	Human Interface Design	**The mobile app may be providing complex information to, and/or requesting data input from, the user.** Limited screen size and the wide variety of platforms available may lead to confusing or misleading information being displayed, or the incorrect information being collected.	Appendix 3
	User Sophistication	**Users will not have been formally trained.** Users may misunderstand the information being displayed or requested and take inappropriate action. Consideration also should be given to whether the mobile app is opened by the user or whether it is running in the background.	Section 4 Section 6.4 (Appendix 1) Section 10.7 (Appendix 5)

Table 2.2: Technical Topics Related to Functionality and Design (continued)

Area	Aspect	Risk Factor	For More Information
Platforms	Platform Hardware	**Mobile app platforms are highly variable.** Many new mobile devices are introduced to the market annually. While there are some small differences in server or PC hardware, they are generally transparent to the operating system and applications. Mobile apps may run on several different hardware designs that can lead to problems with, e.g., operating system compatibility, screen image rendering, and limitations on communication protocols.	Appendix 1
	Platform Software	**A mobile app may have to run on a large number of software platforms.** While the number of operating system options available to mobile users is generally limited (three main options at time of publication), major operating system updates may occur more often than on PCs. Mobile apps are at risk of becoming non-functional as a result of an operating system upgrade. Conversely, some mobile device and/or service providers are reluctant to push operating system upgrades to mobile devices, so a user may be unable to obtain a critical update.	Appendix 1
	Platform Ownership	**Users select their own mobile devices and change them frequently.** Users may replace their mobile device about every two years. Mobile app suppliers have no way of ensuring that the hardware, operating system, and app version all meet expected standards. Users may jailbreak their devices, both increasing vulnerability to hacking and potentially voiding supplier support agreements.	Appendix 1
Platform Limitations	Inadequate Memory	**Mobile medical apps have to compete for devices resources.** Computing resources or storage space may run out because users employ their devices for several purposes, such as music or video.	Section 6.4 (Appendix 1)
	Battery Life	**Users can drain the batteries in their devices.** Where a mobile medical app needs to be running or needs to transmit data, this can present a risk.	Section 6.4 (Appendix 1)
	Emergency Conditions	**The ability of the mobile app provider to address an emergency could be highly dependent on user cooperation.** Where a mobile medical app becomes dangerous to the patient (e.g., due to communications issues, discovery of a software fault needing immediate remediation, the need for an important upgrade or patch), a mechanism (e.g., a notification) is needed to gain user attention and drive the user to accept the appropriate corrective action.	Appendix 1
	Malware Vulnerability	**Effective malware protection is at least partially at the discretion of users.** Mobile devices may not have the same level of protection against hacking or malicious code that is generally available to computers behind a firewall.	Appendix 1

Table 2.2: Technical Topics Related to Functionality and Design (continued)

Area	Aspect	Risk Factor	For More Information
Communications	Interrupted Communications	**Incomplete communications are a much greater risk for mobile apps.** Communication between mobile apps and centralized data handling are at greater risk. This is due to the potential for events as varied as entering an area without a signal, turning off the device, exiting the mobile app, interference from another device, or accepting an incoming phone call.	Section 6.4 (Appendix 1) Appendix 4
	Prevented Communications	**Users can configure their own devices**. Mobile devices are not typically owned and managed by a healthcare provider who understands the risk related to a mobile app. Users can try to reduce cost or increase performance by turning off automatic updates. This could delay or prevent a critical software patch or upgrade, or compromise timely transmission of medical results.	Section 6.4 (Appendix 1)
	Communications Standards	**There are many communication protocols.** These may differ due to national standards, due to the quality of existing cellular infrastructure, or due to the nature of the device selected by the user. Protocols such as Bluetooth or Near Field Communications (NFCs) also need safeguards to reduce risk. The nature and speed of networks can introduce risk related to ensuring data transmissions are complete.	Appendix 4

3 Application of GAMP® 5 to Mobile Apps

The approach described in this Guide is based on the ISPE GAMP® 5 Guide: A Risk-Based Approach to Compliant GxP Computerized Systems [8], which covers compliance and validation of GxP regulated computerized systems. Key concepts which should be considered include:

Product and Process Understanding

A full understanding of the intended use and intended user-base is fundamental to accurately determining mobile app requirements. Such understanding is the basis for quality risk management. See Section 4.

Life Cycle Approach within a Quality Management System

The product life cycle should define the necessary activities to manage the development and support of mobile apps in a systematic way from conception to retirement. See Section 5.

Scalable Life Cycle Activities

Life cycle activities should be scaled according to:

- Impact on patient safety and data integrity
- Complexity and novelty of the app
- Assessment of any suppliers

Science Based Quality Risk Management (QRM)

QRM is a systematic process for the assessment, control, communication, and review of risks. Application of QRM enables effort to be focused on critical aspects of a mobile app. See Section 4.

Leveraging Supplier Involvement

Regulated companies should seek to leverage supplier knowledge, experience, and documentation throughout the life cycle. See Appendix 6.

4 Quality Risk Management for Mobile Apps

This section gives an overview of a suggested QRM approach, based on that described in GAMP® 5 [8] and aligned with the ISO 14971 Standard [11] for medical device risk management. A more detailed description of the approach is provided in Appendix 2.

While mobile applications share many of the same risks that are common to any software, there are many additional risks due to differences in intended use and the operating environment. These include:

- Mobile apps may be required to run on multiple operating systems and hardware platforms in order to reach the desired population. Mobile apps also may be required to run on different operating system version levels.
- The mobile platforms may not be under the control of an of a regulated company's Information Technology (IT) department.
- Connectivity with mobile platforms cannot be guaranteed.
- Users cannot be formally trained.
- User actions cannot be controlled – users can refuse updates, upgrade the operating system, or download other software without the controls normally applied though formal change management.
- Users could cause harm to themselves or others by misusing a mobile app.
- The risk of damage, loss, or misuse of hardware is much higher.
- Limited battery life may lead to data loss.

These and other risks specifically associated with mobile apps are discussed in detail in Appendix 1.

The approach is intended to be applicable to any mobile application with a potential impact on user safety, regardless of whether it is classified as a medical device.

A key aspect in the risk management framework is the intended use, which is defined as the use of a product, process, or service in accordance with the specifications, instructions, and information provided by the manufacturer. This should be considered when assessing mobile applications for public use.

The manufacturer should establish and maintain a process for:

- Identifying hazards associated with the use of a product
- Estimating and evaluating the associated risks
- Controlling these risks
- Monitoring the effectiveness of the control

This process should be documented in:

- Quality management system procedures
- Product design/development process document
- Appropriate plans, e.g., product quality plans, compliance plans

For medical devices, the process may be documented in separate formal risk management plans.

During analysis, the manufacturer should clearly define and describe the intended use, and any reasonably foreseeable misuse, as well as identifying the characteristics (such as functionality, accuracy, reliability, or other aspects) that could affect the safety of users.

Aspects of intended use that may be particularly relevant for mobile applications include:

- Medical/clinical purpose of the system
- User population (e.g., age, visual capability, or other health conditions)
- Platform characteristics
- Application: environment, frequency of use, location, connectivity requirements

The manufacturer should document known or foreseeable hazards associated with the use or misuse of the mobile application in both normal and fault conditions.

When risk analysis and evaluation indicates that risk reduction is required, the manufacturer should apply risk control measures, e.g.:

1. Inherent safety by design
2. Protective measures in the application itself or in the development process
3. Information for safety

These control measures are listed in order of priority and should be documented in specification and design information for the mobile app.

Note: for mobile apps, training and procedural adjustments are not usually an option, because of the limited level of influence over user actions.

The effectiveness of the risk control measures should be verified and the results of the verification should be documented. This can be achieved through development testing and product acceptance testing activities.

An appropriate report or other documented outcome should provide (or refer to) traceability for each hazard requiring control, through to verification of the risk control measures, and an assessment that the residual risk is acceptable.

Information about the performance, reliability, and effectiveness of the mobile app in the market should be gathered by the business process owner and evaluated for relevance to safety.

5 Mobile App Life Cycle

5.1 Overview

Compliance with regulatory requirements and fitness for intended use may be supported by adopting a life cycle approach following good practice as defined in GAMP® 5 [8].

The product life cycle should define the activities necessary for the management of the development and support of mobile apps in a systematic way from conception to retirement. The life cycle of data captured or generated and maintained also should be considered.

The product life cycle described for a regulated company in this Guide should not be confused with the need for a defined approach or method for software development, which is typically the responsibility of the supplier. Appendix 6 describes good practices for suppliers, and these activities perform an important role in supporting regulated company activities.

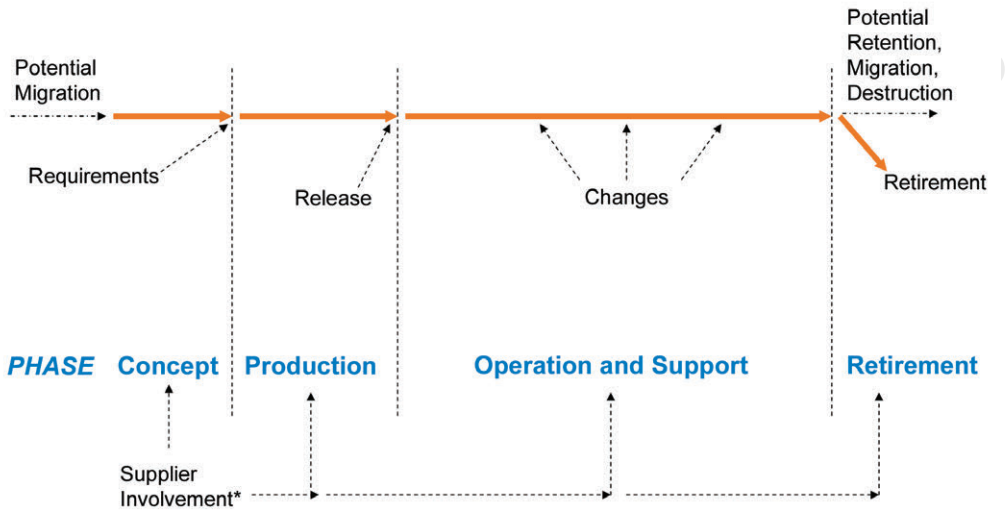
This Guide uses diagrams to represent the life cycle. These diagrams may present relationships in a linear representation. This is not intended to constrain the choice of development methods and models. Suppliers should use the most appropriate methods and models, which may include prototyping techniques.

5.2 Mobile App Product Life Cycle

The mobile app product life cycle is regarded as consisting of four main phases, consistent with those described in GAMP® 5 [8], but renamed to reflect the product life cycle, as shown in Figure 5.1.

- Concept
- Production
- Operation and support
- Retirement

Figure 5.1: Life Cycle Phases



*Supplier may provide knowledge, experience, documentation, and services throughout lifecycle.

The approach described is intended to be aligned with ISO/IEC 62304 [9] and ISO 13485 [10].

During the **concept** phase, an initial risk assessment should be performed (including identifying whether the app is part of a GxP regulated system or a mobile medical app). Initial quality planning also should be performed and initial requirements should be defined.

The **production** phase involves further planning, supplier assessment and selection (where the product is not being developed internally by the regulated company), various levels of functional and design specification, coding, and verification leading to acceptance and release for operation.

Product operation and support is typically the longest phase and can present a significant risk. Data security and integrity, maintaining fitness for intended use and compliance of the mobile app are key aspects. Changes of different impact, scope, and complexity should be managed during this phase. Fault complaints and adverse events should be appropriately managed.

The final phase is **retirement**. This phase involves decisions about data retention, migration, or destruction, and the management and enforcement of these processes.

5.2.1 Concept

For each mobile app, a documented initial risk assessment should be performed. The initial risk assessment should be based on a defined main purpose, intended users, and intended features.

The risk assessment should be performed by appropriate Subject Matter Experts (SMEs), in conjunction with regulated company quality assurance and/or regulatory affairs representatives.

The risk assessment should determine and document whether the proposed mobile app is part of a GxP regulated system or is a mobile medical app. If it is a mobile medical app, the applicable medical device classification should be determined.

Applicable guidance and regulations should be consulted for all jurisdictions where the mobile app is to be deployed, e.g., FDA Mobile Medical Applications [3], and MHRA Guidance on Medical Device Stand-Alone Software [4].

Some applications may consist of both medical device and non-medical device modules or components. GxP regulated medical device modules must comply with the requirements of the applicable regulations. Additional requirements may apply, e.g., in the EU these GxP regulated medical device modules must carry the CE marking. Non-medical device modules are not subject to these requirements.

The manufacturer should clearly identify the boundaries and interfaces between the medical device and non-medical device components, based on their intended use. The manufacturer should ensure that the entire product, including the mobile device or other equipment, is safe and does not adversely affect the performance of the GxP regulated components.

The regulated company for whom a medical device (including a mobile medical app) is licensed bears responsibility for patient safety, but cannot assume liability on behalf of the manufacturer. For example, according to 21 CFR Part 807 [12], a software developer who produces a mobile medical app on behalf of a regulated company (i.e., *"makes a device for or on behalf of a specifications developer or any other person"*) is required to comply with the registration and listing requirements regardless of whether or not they also place the mobile medical app into commercial distribution.

However, a supplier for a regulated mobile app that is not a medical device may not be in a position to evaluate other types of regulatory risk, as this is heavily dependent on a number of factors that may be the responsibility of the regulated company buying the application or possibly a joint responsibility. The regulated company should clarify any supplier expectations, and these should be documented in a contract, a Service Level Agreement (SLA), or a quality

agreement. A product quality plan (or equivalent) should be produced for mobile medical apps and should document the product development process. Responsibilities of regulated companies and any suppliers should be clearly documented.

A product quality plan (or equivalent) can typically include:

- A brief description of the product, including a summary of goals and objectives
- An overview of planning activities, including selection of development method, identification of major tasks, activities, milestones, and related target timings or dependencies
- An outline of roles and responsibilities
- A description of interfaces between groups involved, including interfaces with third parties, e.g., mobile platform suppliers (hardware and software), customers, data providers, app stores
- The quality requirements to be met and procedures to be followed
- A listing of the deliverables associated with the product, e.g.:
    - Product requirements
    - Quality plan
    - Specification and design documentation
    - Traceability matrix
    - Design review documentation
    - Test documentation
    - Risk management documentation
    - User documentation

Note: much of the above will be considered part of the Design History File (DHF)[2] for mobile medical devices, and as such should be managed under formal change control.

The product quality plan should ensure that adequate resources, including appropriate SMEs, are allocated in order to achieve the desired level of documented quality and compliance.

Product requirements should be developed by gathering and defining user needs, intended use, and regulatory requirements including, where applicable, electronic record and signature requirements. Product requirement statements should be complete, consistent, unambiguous, and testable. See Appendix 3 for details of requirements definition for mobile apps.

High-level architecture factors which should be considered when planning for the different environments in which a mobile app may be used include:

- Device connectivity

[2] The DHF is a general regulatory requirement for medical devices. For example, from 21 CFR Part 820.3(e) [13] “Design History File (DHF) means a compilation of records which describes the design history of a finished device.”

- Device components
- Mobile client approach

These architecture factors are described in Appendix 4.

Risks specific to mobile apps may be addressed effectively during planning and requirements definition. Examples of such risks are discussed in Appendix 1, Section 6.2.

Table 5.1 shows how medical device design control requirements, as defined 21 CFR Part 820 [13], may be met by the product life cycle activities described in this Guide. The approach described is also aligned with ISO/IEC 62304 [9] and ISO 13485[3] [10].

Table 5.1: Meeting Medical Device Requirements

Medical Device Requirement	How Met in this Guide
Design and Development Planning	Established product quality plans[4] should describe or reference the design and development activities and define responsibility for implementation.
Design Input	Established product requirements should define the intended use of the product including the needs of the user and patient.
Design Output	Product specifications and design documentation should define and document design output in terms that allow an adequate evaluation of conformance to design input requirements.
Design Review	Documented design reviews should be planned and conducted at appropriate stages of product development.
Design Verification	Product testing should confirm that the design output meets the design input requirements.
Design Validation	Product testing should ensure that products conform to defined user needs and intended uses
Design Transfer	Product testing should ensure that product design is correctly translated into production.
Design Changes	Product change management should ensure identification, documentation, review, approval, and testing of changes
Design History File	Product documentation should be adequate to demonstrate that the design was developed in accordance with the approved product quality plan, and under appropriate design control.

Regulated company Quality Assurance (QA) should perform an independent role in the approval or audit of specification and design activities that may impact public health, patient safety, or product quality.

5.2.2 Production

The development of mobile apps consists of the following activities:

- Finalizing and documenting product requirements, ensuring that they are clear, complete, testable, and unambiguous, including any prototyping and evaluation activities

[3] ISO 13485. This standard outlines QMS expectations that are compliant with EU regulations.
[4] A quality plan is a medical device requirement, e.g., as described in the US regulation 21 CFR Part 820.20(d) [13] or the Australian Therapeutic Goods (Medical Devices) Regulations 2002, section 4.4 [14]. Other regulators have similar requirements.

- Assessment and management of suppliers
- Translating product requirements into:
  - Functional specifications
  - Design specifications
- QRM activities
- Performing design reviews
- Creating and maintaining traceability information
- For mobile medical apps, the creation and maintenance of a DHF
- Producing and managing the code for the mobile app
- Documented software testing – ensuring the developed mobile app is adequately verified
- Developing user documentation, instructions, and training
- Releasing the mobile app for use

These activities are described in further detail in Appendix 5 Production Phase for Mobile Apps and Appendix 6 Supplier Management.

These activities address the design input and design output elements of design control (required for mobile medical apps) by establishing requirements that define the intended use of the product, including the needs of the user, and establishing specification and design documentation that allows an adequate evaluation of conformance to requirements. Based on the medical device classification of a mobile medical app and the specific market regulations that could apply, obtaining premarket approval from one or more regulators may be required at the end of this phase.

Risk management techniques should be applied to identify quality risks and to remove or reduce them to an acceptable level. For mobile medical apps, risks to be evaluated should include potential patient risk, and regulated data integrity. For a mobile app that is classified as a mobile medical device, the supplier is a manufacturer as defined within applicable medical device regulations. This includes, but is not limited to, documentation standards relating to development and support of the mobile app.

Examples of risks associated with the production phase are discussed in Appendix 1, Section 6.3.

5.2.3 Operation and Support

Mobile apps should be supported and maintained in accordance with established procedures. These should cover:

- Change and configuration management
- Incident and problem management

- Product maintenance, updates, and release management
- Data integrity, security, and privacy
- Data backup and recovery

- Data retention, migration, or destruction
- Malware protection (if necessary based on risk)
- Management of distribution channels
- Planning for product retirement and withdrawal
- Medical device registration and listing (for mobile medical apps)
- Medical device post marketing surveillance and medical device reporting (for mobile medical apps)
- Medical device recall, field action and correction (for mobile medical apps)
- Periodic calibration, if required
- Periodic review

The infrastructure supporting mobile apps, such as associated websites and hardware, should be maintained in a state of control and compliance, following established procedures.

Mobile apps should be supported and maintained by appropriately trained, qualified, and experienced staff. For a mobile app, documented evidence of this will normally expected by regulators. Verification of successful installation of mobile medical apps may be required in some regions, e.g., *Medical Information Systems – Guidance for Qualification and Classification of Standalone Software With a Medical Purpose*, Swedish Medical Products Agency [7].

Patient and public data associated with mobile apps and subject to specific regulation should be identified, and should be protected by suitable controls to ensure their security, privacy, and compliance. This should include data managed or hosted by third parties on behalf of a regulated company.

A change and configuration management process should be established to ensure identification, documentation, review, approval, and testing of changes.

When changes are made to a mobile app, sufficient regression analysis and testing should be performed and documented to demonstrate that portions of the software not involved in the change were not adversely impacted. This is in addition to testing that evaluates the correctness of the implemented change(s).

Regression testing also should be performed for upgrades of the operating system of a mobile platform running a mobile app. The testing should verify that the app is compatible with the new operating systems (app stores may remove an app from the store if this is not verified), and that critical functionality is not adversely affected.

Note: the window for such regression testing may be very short for apps not on devices owned by the regulated company, as it is not possible to control when users install the operating system upgrade.

For mobile medical apps, new functions should be reviewed to identify any possible change in use that may lead to a change in device classification or in registration data, or may require re-registration with the health authority.

The range of hardware devices and operating systems upon which the mobile app will be available should be actively monitored and defined and documented during maintenance. Maintenance releases should be tested to ensure that the software continues to work correctly on the supported devices and allowed operating system versions.

An incident management process should be established to receive and address problems, incidents, comments, or complaints related to a mobile app, received either from users or other sources. This should coordinate with a problem identification and resolution process to ensure identification, documentation, and resolution of problems attributable to correctable faults with the mobile app or mobile service provider.

User feedback mechanisms should be provided from within a mobile app, where possible. This may be via a form that is submitted to a web service, or mail-based feedback. Processes should be established to deal with feedback. Failure to provide an easy method of feedback or failure to respond to the feedback in a timely fashion may result in negative public evaluation of the mobile app (e.g., via social media). Users may resort to unintended channels, e.g., a regulated company's customer complaint line. This may introduce a high administrative burden that may be unhelpful or not cost effective.

Customer feedback may include a variety of topics including:

- Questions
- Compliments
- Incident reports
- Complaints
- Adverse events

Mechanisms should be established to deal appropriately with each. Local legal requirements may demand that feedback mechanisms are available in the local language.

Complaints regarding mobile medical apps should be managed according to established processes, which include medical device reporting requirements.

Provisions should be made to interface **any** customer feedback channels that allow free text with adverse event reporting systems; and ensure adequate review and analysis, e.g., part of the post-marketing surveillance and reporting process.

The content of a mobile app can become out-of-date. Planning for maintenance should include consideration of the need for regular assessments and possible updates of mobile app content to ensure that this content is still current and valid.

Prior to release of new product versions or upgrades containing new functionality, the initial risk assessment should be reviewed to ensure that the conclusions remain valid for the specific release in question. Software patches or updates should be evaluated to determine what should be pushed to user devices and what can wait for users to pull.

Examples of risks associated with the operation and support phase are discussed in Appendix 1.

For further information on operation and support of systems, see GAMP® 5 [8].

5.2.4 *Retirement of Mobile Apps*

The retirement, replacement, or withdrawal of mobile apps should be performed in accordance with an established process or plan. This plan should take into account potential risk to users should they lose access to the mobile app. Any technical measures required for retirement (e.g., disabling the app following a failed handshake protocol) should be addressed at the design state.

The retention, migration, or destruction of data associated with a mobile app product should be performed in accordance with an established process or plan, taking into account applicable regulatory and other legal record retention requirements.

Users should be provided with sufficient notice of retirement of a mobile app or version. The right to retire and the required prior notification should be covered in the terms and conditions for the mobile app. Any expected supplier actions associated with mobile app retirement (e.g., transfer of database content to the regulated company) also should be defined.

The mobile app should be designed in a way that allows for its retirement either forcibly or automatically such as in the event the application is no longer supported.

Examples of risks associated with the retirement phase are discussed in Appendix 1.

5.3 Mobile App Data Life Cycle

Data management can be challenging when some of the data that needs to be managed may originate and/or reside on a device outside the regulated company's direct control. Data management should be considered during requirements planning with potential actions identified, defined, and documented.

5.3.1 Data Management during Mobile App Development

When planning for mobile app development, regulated companies should consider that they may need to manage data about customers and/or patients. Depending on the nature of the data, this carries significant regulatory and legal responsibility. In most jurisdictions, there are general data privacy laws, and in some jurisdictions, there also may be specific health care data privacy laws. It is recommended that only data needed directly to achieve the intended use is collected.

The following should be considered:

- **What data will be collected and how will it be stored**? This includes data required prior to downloading/ installing the mobile app as well as data collected through the mobile app.

- **How will the data be used?** Any data not required to achieve the intended use of the mobile app should not be collected, or should be collected via a different route.

- **Who will have access to the data, and why?** Taking into consideration that data collected may be subject to national/local privacy laws, any use for reasons other than direct support of the mobile app or the wellbeing of the user should be carefully considered, and may need to be vetted by the appropriate function, e.g., the corporate privacy office or the data protection officer.

- **Retention period for the data:** data should be retained in compliance with corporate record retention requirements, taking into consideration that regulations or laws related to medical records may be applicable. If data resides on the user's mobile device and needs to be deleted, mechanisms should be designed both to enable and to enforce this. Complete data destruction may not be possible without wiping the entire device.

- **User ability to access, view, transfer, or delete data**: the regulated company should have a data management plan that is customized for the information collected related to a mobile app. The types of considerations can be varied, e.g.:

  - **Some records that the regulated company collects may have to be retained regardless of a user's desire to delete it.** Policies governing such situations should be established prior to release of the mobile app. These policies should be part of the license acceptance agreement or part of the informed consent, e.g., for a clinical trial.

  - **Some other records (e.g., Electronic Health Records (EHR)) need to be transferrable or capable of being copied** so that health records can be made available should the user change physicians.

- **Some records may be user managed.** If there are no laws or regulations governing the records and there is no corporate liability concern, it may be sensible to allow the user the capability to manage their own records.

- **When designing a mobile app, consideration should be given to whether records need to be viewable by the user.** In some regions, this could be a legal requirement.

- **Disaster recovery/business continuity measures:** depending on the nature of the mobile app, it may be critical to patient safety to ensure that the mobile app is available when needed. In some cases, this may mean that remote back-up of records that reside on the mobile device is an appropriate protection. The means to do so, and to restore them in a timely manner, should be defined.

The Recovery Time Objective (RTO) and Recovery Point Objective (RPO) define how quickly the mobile app needs to be recovered and how much data loss can be tolerated resulting from a disaster. Both should be defined, based on risk, in order to select the appropriate recovery strategy, e.g., a local buffering mechanism may be appropriate.

- **End of life planning:** the regulated company should understand how to approach the following scenarios related to record retirement:

  - **What needs to be done with collected data when a user stops using the mobile app?** Data retention considerations should consider how long the regulated company is willing to retain non-regulated data, such as some configuration settings.

  - **How will data be destroyed at the end of the retention period?** This could conceivably impact replication strategy if copies of records that were deleted from a database at the end of the retention period are subsequently re-downloaded from the mobile device.

  - **What will happen to the data when the regulated company retires the mobile app?** This needs to include consideration of users who continue to use the app after retirement, and should be coupled with other retirement strategies.

  - **End of life for mobile device:** in addition to planning for the retirement of the mobile app, provisions need to be made for retirement of a mobile device owned by a user who wishes to continue to use the mobile app. If data transfer between devices is required, a mechanism for this should be developed and validated.

  - **What will happen when an employee leaves or changes role?** Company policies and procedures should cover the return and reuse of devices holding sensitive data. Particular attention should be given to the removal of data from employee owned devices.

5.3.2 Data Management in the Operational Phase

Prior to the beginning of the operational phase of the system life cycle, the data management approach defined during the requirements phase should be implemented so that the transition to active data management is seamless. As soon as the mobile app is released to the public, the regulated company should begin managing any data collected by the mobile app.

Appropriate measures to protect information stored on servers or on the mobile device should be established and working, considering that information collected could be subject to legal data privacy protections, EHR requirements, or GxP regulations. This includes:

- Appropriate disaster recovery/business continuity protections. This is an issue that could be of paramount importance for some medical device uses.

- Appropriate user access controls, including active management of who can access any records stored on servers

- Appropriate controls to prohibit unauthorized access, possibly including antivirus, encryption, and intrusion detection

If personal data is collected by a mobile app and later stored to a centralized database, it should be possible to remove the data. Depending on local laws, this may have to be on request or achieved automatically.

Automated processes (e.g., back-up of data from user devices, if applicable) should be monitored to ensure proper function. Such automated processes should be verified during validation.

5.3.3 Data Management for Retirement

Part of managing the retirement of a mobile app is placing records pertaining to the production and operational phases into an appropriate inventory that will support record destruction at the end of the retention period. Such a record inventory should be able to support regulatory inspection of the records throughout that retention period. It also includes all of the controls appropriate to ensure the protection of the integrity and confidentiality of the records. It should be compliant with data privacy, as well as GxP record management requirements.

6 Appendix 1 – Mobile Apps Risk Landscape

6.1 Overview

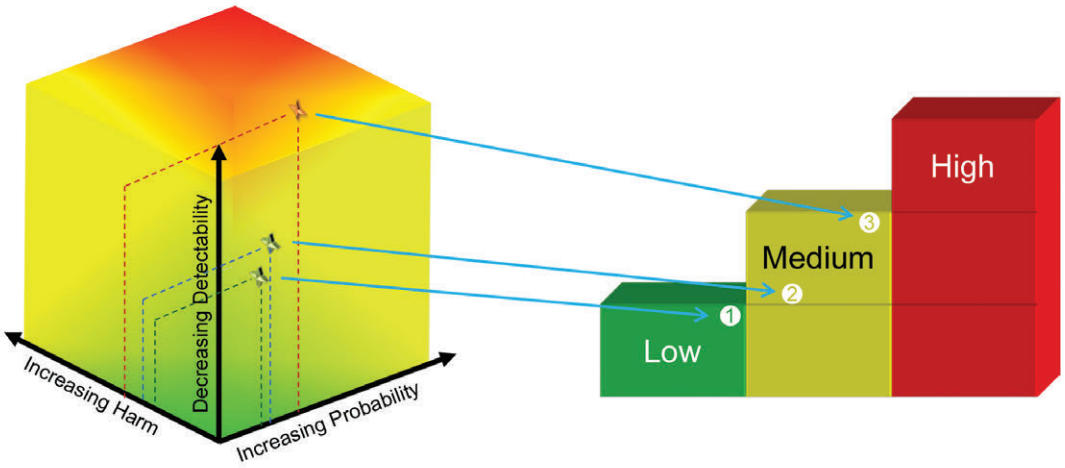
Mobile apps share many of the risks associated with traditional server or PC-based applications; however, the nature of the mobile platform and environment may require specific controls to manage risks specific to mobile apps. This appendix presents the risk landscape with the intent of clarifying these new or different risks.

Risks at all life cycle phases should be understood and risk mitigation should originate in the planning phase. This allows the design of product or process to be adjusted to either avoid risk or reduce risk to acceptable levels, both economically and effectively. Recognizing in the planning stage that risks are deemed too severe to continue avoids the investment of time and resources in an untenable project.

While mobile apps share many of the same risks that are common to any software, there are many additional risks due to significant differences in the operating environment.

Risk should be considered as a continuum (see Figure 6.1). Generally risks that are not addressed at an early stage (usually during planning), do not manifest until a later life cycle stage (typically during operation, or retirement). This section addresses risks where they most commonly could have the greatest impact; however, when considering how the risk could be avoided, mitigation should be undertaken during the earliest phase possible.

Figure 6.1: Risk as a Continuum, Not a Series of Steps



The continuum on the left of the Figure 6.1 shows risks 1 and 2 are clearly closer in terms of potential harm, probability, and detectability than are 2 and 3, yet with the view approach on the right it is likely that 2 would be assigned controls that might be overkill in order that all "medium risks" are approached in the same way. Similarly 3, as a very elevated medium risk, might need some (but not all) controls that are typically reserved for high risk.

Depending on the nature of the regulated mobile application, there may be medical device regulations that must be considered, especially as affects the operational phase of the life cycle. Classification as a medical device has the potential to significantly complicate the risk landscape. Some medical device controls may be in conflict with other laws and regulations (e.g., reporting requirements vs privacy law), and in such cases, the whole concept of whether the application is appropriate to place on a phone or tablet, and whether it should be restricted to specific markets, should be evaluated.

The following sections consider specific risks associated with the Concept Phase, Production Phase, Operation and Support Phase, and Retirement Phase.

6.2 Concept Phase Risks

Many risks can be quickly addressed when defining the system, and appropriate product requirements may mitigate or avoid numerous risks. For example, many risks that become obvious in the operational phase can be mitigated (at least partially) during the planning and design of the mobile app. Conversely, failure to consider these risks at the earliest opportunity could result in significant expense later in the development life cycle, or result in a decision that the application cannot be deployed. Table 6.1 lists examples of risks that should be addressed during the concept phase. When evaluating risks in later phases; however, thought should always be given to the question, "Could this risk be avoided by adjusting the user requirements prior to design and build?"

Considerations during the Concept Phase include:

- **Classification as a Medical Device:** if the mobile app is used to monitor or control factors related directly to patient health (e.g., a smartphone as a wireless interface transmitting blood sugar readings from a blood glucose meter to a database referenced by a physician), it is a medical device and it is subject to the controls of relevant health authorities for medical devices. Where the classification as a medical device is unclear, appropriately knowledgeable SMEs should be involved in the classification. The SMEs should be familiar with the regulatory requirements of the market(s) where the product will be released.

- **Medical device controls** can be significant and vary based on the classification of the device. Business process owners of the mobile app may not understand the resulting requirements for documentation and management of the mobile app. Some mobile medical apps may require pre-market approval before they can be released to the public.

- **Security:** security issues would usually be most relevant for mobile medical apps classified as medical devices, but also can apply to other issues, e.g., privacy law (see NIST Guidelines [15]).

- **Hacking:** if the potential consequences of hacking are sufficiently serious, protection should be planned during the Concept Phase, e.g., firewalled mobile web architecture, data storage in a secure cloud.

- **Network considerations:** if the mobile app includes communication to a physician, the potential impact of imperfect transmission of data may be serious. If the mobile app will allow a physician to adjust treatment remotely, it is higher still.

- **Other communication channels:** the selection of a short-range communication technology, e.g., Bluetooth or Near Field Communications, could have an effect on security.

- **Privacy considerations:** if the application will contain sensitive personally identifiable information (such as relating to disease state), the need for encryption or some other mode of protecting this information should be considered. The entire mobile device should be taken into account. For example, medical information keyed to a patient ID number may not be significant without the key linking that number with a patient name, but the same information paired with a smartphone address book or calendar application could link a disease state to an individual.

The basic decision about what architecture model to follow can be affected by all of these considerations. It may be easier to protect sensitive information that is stored centrally rather than on thousands of privately owned devices. For example, a mobile web approach with secure cloud storage of the data could a better solution than a device-centric approach.

The nature of the mobile app may be a factor in deciding what platform(s) to target for development. If the mobile device is to be part of a larger medical device, it may make sense to develop on only one platform that appears to be the best technical choice. However, if the mobile app is intended to be accessible to a wide variety of customers, the developer may need to plan to support the mobile app on multiple operating systems in multiple versions, and on multiple devices. This choice will give rise to several operational risks (see Section 6.4).

Table 6.1: Examples of Risks Encountered during the Concept Phase

Issue Type	Hazard	Risk
Technical	Inappropriate choice of architecture (e.g., mobile app versus mobile web decision)	Inadequate design of security and user interface
	Use of short range communication	Short range communications provide another avenue for access and related security risks
	Incorrect use or unexpected failure of the mobile app	Adverse impact on patient health
	Inadequate security	Security breach allowing patient data to be accessed, modified, lost, or stolen
	Unsuitable data storage approach	Inadvertent data loss due to user action (e.g., deletion of data while attempting to restart a non-functioning mobile app)
	Inadequate network communications strategy	Loss of data during communication failure (e.g., due to lack of local buffering)
Supplier	Inadequate development practices	Delivered product does not meet functional, quality, and/or documentation requirements
	Inadequate support practices	Malfunction of, or inability to use, the mobile app
	Supplier lacks understanding of legal and regulatory environment	Mobile app does not comply with regulatory requirements
Regulatory and Legal	Lack of understanding of pertinent regulations, including medical device classification and the potential need for premarket filing	Failure to recognize a mobile app as a medical device could delay product launch or force a market withdrawal until filing is properly completed
	Unclear regulatory accountability	Failure to comply with regulatory requirements
	Inadequate data privacy policies	Increased risk of data privacy breaches

6.3 Production Phase Risks

Most of the risk in this phase relates to the supplier who actually builds the mobile app. Table 6.2 summarizes production phase risk considerations.

While there are some large, mature companies that do mobile app development, a large proportion of the developer community are smaller companies. A mobile app sponsor may wish to engage a small company with few employees (e.g., a garage developer) or perhaps a small department in a non-regulated company (e.g., an advertising agency). The level and quality of documentation that can be expected from such a development company is likely to be less than pharmaceutical or medical device companies typically demand for regulated software. Very small companies may not see the value of generating such documentation, but even if they are willing to generate it, it may add significantly to the cost of the software. When bidding a job out to small development companies minimum documentation requirements should be clearly defined in the contract or SLA, so that costs and expectations are understood by both parties. The regulatory risk related to inadequate software development documentation is considerably higher for medical devices. The need to maintain design history records for devices extends this requirement (and risk) into the operational phase.

Small software development companies may be less financially stable. If an mobile app is intended to be used for several years, special measures may be necessary to preserve software support services should the developer company fail, be taken over, or choose not to provide further support services.

Even with a software development organization that employs 10 to 20 people, customers should consider that smaller companies may not have a mature QMS comparable to that of large software developers. Even in situations where smaller organizations do have a QMS, they may not have sufficient staff to allow for full separation of roles, particularly in relation to an independent QA function. It may be that responsibility for software QA could be assigned to the Head of Software Development simply because that individual has most knowledge and experience. Such a conflict of interest can be problematic for regulated companies and should be avoided.

A regulated company should carefully assess potential suppliers in order to make their own expectations understood to the supplier and to understand the capabilities of the supplier, as well as their willingness to adapt processes and practices to meet customer needs. Sponsoring regulated companies should not use suppliers who are unable to meet minimum standards, unless compensating controls to guarantee quality are established. It is the responsibility of the sponsoring company to ensure that these controls are adequate.

Table 6.2: Risks Encountered during the Production Phase

Issue Type	Hazard	Risk
Technical	Use of multiple platforms (e.g., Android®, iOS®, Windows®, and many hardware suppliers)	Mobile app may not perform as intended across all combinations of platforms and documentation may not reflect all possible configurations
	The rapid turnover of devices in the mobile environment means that it is likely that several versions of each supported operating system and many devices, all slightly different, need to be supported	Appropriate backward compatibility is not maintained which could lead to failures in mobile app performance
	Information display and data entry limitations, e.g., screen size or resolution, not adequately considered during design	User errors due to misunderstanding or misinterpretation
Supplier	Use of inappropriate or not suitably qualified or experienced developers	Mobile app is developed in an informal manner, without the rigor of a formal QMS, leading to the app not being fit for purpose
	Insufficient staffing impacting quality-related decision making	Mobile app is developed and released without appropriate assurance that it meets the defined requirements
	Inadequate documentation management	Formal, controlled, documentation is not available to verify and demonstrate that the mobile app is fit for intended use
Regulatory and Legal	Lack of understanding of medical device regulations	Companies that have no history in the medical device industry are at risk of failing to comply with regulations and standards, if not given significant support in creating and maintaining medical device documentation (e.g., the DHF)
	Lack of understanding of applicable laws and regulations (e.g., GxP, PDMA compliance, Data Privacy)	Failure to comply with laws and regulations putting companies at risk of regulatory action
	Inadequate knowledge of applicable regulations relating to electronic records and signatures	Mobile app does not provide the technical controls required to enable compliance with electronic record and signature requirements in those markets where regulations allow for the use of such records and signatures

6.4 Operational Phase Risks

The Operational Phase of the life cycle of a mobile app can present the most opportunity for problems. Some of these are specific to particular platforms and/or operating systems; some are attributable to the ideas of suppliers and some will be due to user actions.

Some regulated companies may use a QA function strategy in which the company's employees run company mobile apps on their own mobile devices. There may be legal implications that differ from those employees who have access to company provided mobile devices, such as the company's right to remote-wipe an employee's mobile device.

Table 6.3 summarizes operational risks.

6.4.1 Technical Risks

Platform support is a decision that should be made at the Concept Phase, but which can have operational implications. It is easier to support an application that runs on only one hardware/operating system combination, but this may not be an effective approach to reach the desired user community. Where a single platform is used, many of the considerations described in this Guide could still present risk because of the fluidity of the mobile environment.

Hardware and operating systems change rapidly, making the landscape significantly less stable than the platforms that have traditionally been used for regulated applications. Manufacturers may introduce several new devices annually. These devices may be incremental developments, e.g., introducing faster communications (e.g., 3G to 4G), new modes of interaction (e.g., NFC), or a different Central Processing Unit (CPU); or they may be evolutionary, introducing a better screen, more memory or storage, or perhaps an existing CPU clocked at a higher speed.

The speed at which suppliers introduce new equipment makes them reluctant to update existing models, e.g., when the next version of the operating system is launched. A factor that complicates the ability to stay current is the supplier practice of customization of the operating system. This can slow the update process, because the handset supplier has to modify the operating system code to retain their branding. Ultimately, mobile app developers may need to support several versions of the various mobile operating systems or risk a failure of the application.

The customization of an operating system by both Original Equipment Manufacturer (OEM), suppliers, and by mobile service provider introduces another major challenge, as it is reasonably common for mobile apps not to function as well (or in some cases at all) on a subset of the platforms using a particular version of an operating system. Supporting several major mobile operating systems can present a challenge, i.e., performing adequate regression testing when changes are made.

Depending on the nature of the mobile app, this could represent a risk to patients. Understanding this, companies could find it necessary to support their mobile app on only a subset of operating system versions and platforms. In some cases, companies may want to disable the software on platforms which they no longer intend support. This may require a function built into the software to prevent use on inappropriate platforms (the "poison pill" strategy). This approach could prevent a mobile app from running without a "handshake" to a centralized server. Such a strategy could be used to force an upgrade or to disable the mobile app.

Note: the handshake strategy also has risks, as the mobile app could be non-functional due to communications issues outside of the user's control, e.g., a weak or non-existent signal. Where there is a risk to patient safety if the application is unable to handshake, there should be a temporary override mechanism. This approach should be carefully planned and designed during development, and reflected in user terms and conditions, licenses, or contracts. The approach should consider the relative risks related to the use of an outdated mobile app versus the sudden withdrawal of the mobile app.

Operating systems are vulnerable to malware to some extent. Mobile app developers should understand the relative risks and impact of this vulnerability. Two impacts of compromise due to malware are:

1. Loss or corruption of data

2. Theft of sensitive information

If risk is deemed significant, various solutions could include:

- Requiring the installation of a mobile security application
- Encryption of data (at rest and/or in transit)
- Replicating data to a remote server
- Not storing information on the device

Change control can be challenging. Wireless providers generally want to keep users of each device at a specific operating system version in order to minimize complexity of support and they roll out new software downloads at their convenience. Ensuring that a small percentage of mobile apps work properly on the new version may not be a priority.

Changes to a device when users download new software are usually inevitable. Mobile devices generally do not have the capacity to back out of changes that have undesired consequences. This risk can be mitigated by careful design and testing of mobile apps.

The mobile app provider also may want to drive changes. If a change is considered mandatory, the "poison pill" strategy may be appropriate, because some users may have devices which are set to refuse automatic upgrades. This approach should be reflected in user terms and conditions, licenses, or contracts.

Careful consideration should be given to adopting configurations that could be altered by a standard change, update of the operating system, or of an unrelated mobile app.

6.4.2 Supplier Risk

The diversity of possible suppliers involved during operation can lead to a range of risks being introduced. The loss of signal can present a technical, supplier, or user risk and could result from a:

- Cellular service provider problem
- User entering a building or structure that blocks the signal
- User entering a "dead zone"

A similar issue could arise if the user moves from an area with high speed 4G service to one with 3G or lower service. Caching information for later transmission could be a solution, but this could be inadequate to address some medical emergencies.

The variety of global wireless standards presents another challenge. For example, North American devices may not work on European networks, and vice versa. Pre-market research should verify that communications standards are understood and addressed for all regions where the device is intended to be used.

Risks related to having an adequate signal when it is needed should be understood and assessed. Depending on the risk related to loss or inadequate signal, it may be desirable to have the mobile app display an alarm or trigger an audio alert where communications are inadequate for proper function.

Following core technical standards and development methods, e.g., from Apple®, Google®, or Microsoft®, may help to mitigate the challenge of supporting multiple devices and versions.

Direct user support is usually provided by the company marketing a mobile app. The group developing the mobile app, e.g., a small developer company may not be equipped to provide ongoing user support via, e.g., a helpdesk. If no helpdesk is provided for a mobile medical app, patients may find other ways of asking questions, such as:

- Through customer complaint channels of the sponsoring company:
  - May provide the answers that customers need, but introduce a high administrative burden that may be unhelpful or not cost effective.
- Through calls to the wireless provider:
  - May not provide any answers specific to the mobile app, may cause annoyance and possible risk to users.
- Via websites, social media, or ad-hoc web searches:
  - May be effective, but is unlikely to be authoritative and may lead to increased patient risk due to incomplete understanding of the mobile app.
- For a mobile medical device, through a call to the medical practitioner:
  - Medical practitioners may need to be given special training and have access to second level troubleshooting support.

The need for a help desk and the risks associated with the decision whether to provide one should be carefully evaluated.

Note: the support function for a mobile app that is classified as a medical device becomes route for adverse event reporting with associated requirements for reporting, secure electronic record maintenance training, etc.

6.4.3 User Risk

A significant source of variability in the risk landscape is user actions, as these are not controlled. Users may be non-technical although for some mobile apps, this may be less of an issue. Mobile apps used by clinicians may be lower risk because these users may be more aware of risks to patient safety. Conversely, the impact of improper use, e.g., of mobile medical app that controls a syringe pump infuser, may have a much higher potential for harm.

User actions may present a risk to the effectiveness of a mobile app, e.g.:

- **Control over undesired software updates:** users may unintentionally allow an update to bring a mobile app out of a supported environment, e.g., by:
  - Accepting an operating system update before the mobile app supplier issues a supported version
  - Accepting a mobile app upgrade that is not supported on the current operating system

  These risks can be mitigated, e.g., within a clinical trial, by providing mobile devices that are “security wrapped” and locked down to corporate standards, although jailbreaking (see below) may be an issue. The risk of accepting inappropriate mobile app updates also can be mitigated by technical checks that prevent installation on an unsupported operating system.
- **Control over desired software updates:** ideally, the user’s device should be configured to accept updates to the mobile app automatically although accepting all updates automatically involves risks. It may be preferable to encourage users to turn off automatic updates and to download updates through system notifications, email, or SMS; however, the user has ultimate control over whether updates are accepted and installed. The use of “security-wrapped” mobile devices may provide mitigation.

- **Potential software conflict:** users may download unrelated software without the controls normally applied though formal change management leading to potential conflicts, e.g., negative interaction with the mobile medical app functionality, competition for computing resources.

- **Mobile device damage:** mobile devices, especially smartphones, may be mishandled and damaged and this may affect hardware performance.

- **Inadequate hardware resources (e.g., memory or storage):** the internal storage memory of mobile devices may become filled with user files (e.g., music). If a mobile app needs to store data, this can present a risk. Where software resides in memory, and could (in theory) inhibit a memory-intensive medical application. Where it is possible to segregate disk space and memory for the exclusive use of the mobile app, this could provide a beneficial technical mitigation.

- **Battery capacity:** demands on battery power can lead to rapid draining. If a mobile medical app needs to record and/or transmit health related data constantly, this could become problematic.

- **Unlocking:** this is the practice of breaking security protections to bypass supplier restrictions on the use of a mobile device, usually for the purpose of using it on a competitor's network. While this is probably less risky than "jailbreaking," it could lead to communication or compatibility issues.

- **Jailbreaking:** "jailbreaking" is the disabling of provider security controls to allow access to the root operating system, and permits the user to load apps not approved by the operating system supplier. It makes the device more vulnerable to both malware and to user-attributable malfunction, and typically voids the warranty.

- **Travelling:** users with mobile medical apps that have patient safety implications should be made aware of the risk of travel to countries where the mobile medical app is not supported. This should include notification of regional limitations with system documentation, as well providing support for traveler questions. In addition, users should be made aware that there may be device limitations (e.g., incompatible networks) or contract issues (e.g., the user's contract does not support international communication). The help desk or other support staff should support patients by providing relevant information.

- **User-Controlled Configuration:** the user may be able to adjust settings on the mobile device that affect the performance of the mobile app, e.g.:

  - The user may configure the device to accept updates only when connected to a wireless LAN, which could affect the supplier's ability to distribute a critical software update.

  - Locking the device display in portrait mode could prevent the user from seeing critical data that is visible only in landscape mode.

  These risks could be addressed using technical controls that override the settings, but the risks should be recognized in the Concept Phase and acted upon in the Production Phase.

User reaction to mobile app failure also may be a source of risk. Users may dismiss or ignore error messages, or may take ineffective or damaging actions. Users may continue using a malfunctioning mobile app without understanding the potential risk. Deleting and reloading a malfunctioning mobile app may work for a productivity application, but could make the problem worse if it erases stored medical data. If appropriate, warnings should be part of the uninstall program.

6.4.4 Regulatory Risk

Regulatory risk during the Operational Phase can involve several aspects of regulation. Data may be subject to Electronic Record and Signature (ER&S) regulations, such as US FDA 21 CFR Part 11 [16], EU GMP Chapter 4 [17] and Annex 11 [18], medical device quality regulations, or other GxP regulations.

The following should be evaluated:

- **ER&S Requirements:** if the mobile app maintains electronic records or signatures they should be maintained through the Operational Phase in accordance with the relevant ER&S regulations. For further details, see GAMP Good Practice Guide: A Risk-Based Approach to Compliant Electronic Records and Signatures [19].

- **User Access:** the risks should be evaluated relative to unauthorized access to the mobile application. Although it is not generally considered an ideal user experience, it may be appropriate to require authentication to access the mobile app. This should not be configurable; offering users an option to bypass this security could put the developer and the company marketing the application at risk. Controls commensurate with the level of the risk should be established. If the mobile application controls dosage, e.g., the patient could be harmed from unauthorized use of the mobile app. If there is a chance that critical data could be compromised, more rigorous user access controls (e.g., biometrics) may be appropriate.

- **HIPAA and Similar Regulations:** if the device is creating EHRs, laws such as the US Health Insurance Portability and Accountability Act (HIPAA) may have requirements that may apply. In particular, HIPAA addresses protection of sensitive Personally Identifiable Information (PII) through means such as encryption.

- **Data Privacy:** if patient data is accessible via a clinician's mobile device, controlled access and encryption are probably both indicated. National laws protecting privacy vary greatly from nation to nation, and even from state to state in the USA. Data privacy should be considered in light of aggregated data; while the mobile app may be below a threshold required for data privacy controls, it is possible that when considered together with an address book application there is sufficient connection between sensitive information and personal information to trigger the need for a control such as encryption. Failure to comply with data privacy rules can result in massive legal penalties, and data breaches with concomitant loss of personal information can be hugely damaging both financially and for a (regulated) company's reputation. Privacy protection laws vary greatly, so local legal and regulatory advice relating to privacy requirements should be obtained from each market where the mobile app is likely to be deployed.

- **Medical Device Records:** if the mobile app is classified as a medical device, there are specific rules for the documentation that must be created and maintained for the design, development, and testing of the mobile app. A DHF should capture all medical device records created for the mobile app.

- **Field Alerts and Recalls:** regulations require that regulated companies have the ability to recall medical devices that are defective or fail to meet requirements. However, in some markets, it may be illegal to remove software from a mobile device without permission. It may be possible to address this in a software user license agreement. A strategy for notification of users if there is a problem also should be considered, and depending on the nature of the device, it may be necessary to embed a "poison pill" in the software that can disable the program if patient safety would be threatened by further use. However, this may still be dependent on factors such as the user not blocking updates, available communication channels, etc.

- **Mobile Apps that are not Medical Devices:** there is significant interest in using mobile platforms like tablets to provide information related to manufacturing processes and as input devices for process control systems. The operation of such devices would need to comply with GMP requirements in accordance with the regulated company's policies and procedures. Similarly, mobile devices have been used in clinical studies for the collection/reporting of clinical data. This type of use would require meeting GCP regulatory expectations, again in accordance with the regulated company's policies and procedures.

Table 6.3: Risks Encountered during the Operational Phase

Issue Type	Hazard	Risk
Technical	Mobile device manufacturers may customize the operating system (e.g., Android®) to provide a distinct user experience to their device	Mobile app may not perform as intended on the customized version of the operating system
	Updated/new hardware	Compatibility issues due to variations in hardware (e.g., screen geometry, resolution, and size; physical keyboards; soft versus hard buttons)
	Inadequate security/access control /malware protection	Security breach allowing patient data to be accessed, modified, lost, or stolen
	operating system updates/patching	Update leads to incompatibility with mobile app functionality
	Application failure	Range of unintended consequences including failure to operate, functionality failure, and data integrity issues
	Inadequate computing resources (e.g., memory)	Risk of failure, e.g., due to timeout functions, actions driven by user frustration or loss of data
	Failure to consider actions subsequent to a failed change	Unable to revert non-performing mobile app to a previous version
	Reliability of data transmission	Interruption of transmission leading to data loss or corruption
Supplier	Lack of support for operating system updates on older mobile devices	Incompatibility issues
	Network reliability (e.g., poor network coverage)	Interruption of transmission leading to data loss or corruption
	No availability of 3G/4G network	If a device depends on dense data transmissions, the loss of 4G coverage, even if 3G takes over, could impact the usability of the device (leading the user to close the mobile app) or possibly affect data integrity
	Incompatible mobile standards (e.g., GSM, CDMA, LTE)	Lack of required standard at operating location affecting performance of mobile medical app
	Planned obsolescence of mobile hardware	Incompatibility issues
	Lack of appropriate help desk support	May lead to inappropriate use of the mobile app by users. The help desk may become an undesirable new channel for complaints and adverse event reports
	Lack of formal arrangements for integrated service support	Calls made to service provider are not redirected to the regulated company

Table 6.3: Risks Encountered during the Operational Phase (continued)

Issue Type	Hazard	Risk
User Risks	Lack of control over operating system updates	Users may either: • Decline an operating system upgrade required for correct operation of the mobile app • Install an operating system upgrade that the regulated company does not yet support
	Lack of control over mobile app updates	Users may either: • Decline required mobile app updates • Install a mobile app update which is not yet supported by the regulated company on a particular operating system version
	Uncontrolled downloads of unrelated software	Can lead to negative interaction with mobile medical app functionality or competition for computing resources.
	Complex user interface	Improper use of the mobile app (may be more likely for mobile apps used directly by patients than for a healthcare practitioner)
	Uncontrolled user actions	Could result in: • Failure to address device damage, possibly affecting performance • Misuse of the device
	User-control of configuration settings	Mobile device configuration may be compromised leading to failure of mobile app
	Ability to jailbreak/hack device	Compromises the integrity of the mobile app (e.g., by installing unsupported software, and/or integrity of data)
	Inadequate battery charge	Interruption of functionality or data transmission leading to data loss or corruption
	Running out of storage space on the device	Possible data loss
	Travelling to an area where the app is not supported	Issues with functionality due to connectivity, device limitations, or contract limitations
Regulatory and Legal	Inability to recall mobile medical app	Faulty mobile medical apps available in market place, and failure to comply with regulatory requirements Legal implications related to trying to withdraw a locally installed defective medical device as some jurisdictions may restrict the ability to withdraw or disable software
	Inadequate access control	Unintended use by an unauthorized user allowing patient data to be accessed, modified, lost, or stolen

Table 6.3: Risks Encountered during the Operational Phase (continued)

Issue Type	Hazard	Risk
Regulatory and Legal (continued)	Inadequate operational change control	Failure to maintain design and other specification and verification records (e.g., a design history file) in accordance with applicable regulatory requirements
	Lack of understanding of applicable laws and regulations (e.g., GxP, PDMA compliance, data privacy, electronic health records)	Failure to comply with laws and regulations putting companies at risk of regulatory action
	Inadequate knowledge of applicable regulations relating to electronic records and signatures	Mobile app does not provide the technical controls required to enable compliance with electronic record and signature requirements in those markets where regulations allow for the use of such records and signatures
	Lack of understanding of applicable requirements for electronic health records	Failure to comply with requirements for confidentiality and portability (e.g., HIPAA in the USA)
	Lost or stolen device	Loss of personal data may compromise compliance with applicable data privacy laws
	Use of BYOD strategy (mobile app functioning on an uncontrolled mobile device)	Mobile app performance is compromised. Legal implications surrounding the company's right to remote-wipe employee's mobile devices

6.5 Retirement Phase Risks

There are complications and risks associated with withdrawing a regulated mobile app from the market. Table 6.4 provides a summary.

Software can be run as long as the supporting platform remains viable. This is not an indefinite period, but it is unlikely to coincide with a desired withdrawal date.

Where the software architecture employs a mobile web component, retirement may be simpler because that component of the functionality can be turned off, possibly after a period of read-only access. If the application resides solely on the mobile device; however, it becomes more difficult to control. There may be legal obstacles to removing the software from a user's device. If users have automatic updates disabled that can add a technical complication.

The mobile app should be designed to allow for its retirement, either manually or automatically, such as in the event the application is no longer supported. Typically, when the mobile device is connected to the network via cellular connection it is possible to ensure that the mobile app is updated or retired. When the mobile device is not connected to the network; however, the upgrade or retirement signal is not received by the application and this can present a risk.

An embedded timer within the mobile app can provide a method to ensure that the mobile app stays current. The purpose of the timer is to notify the user that they need to connect to the network periodically to verify that the version of the mobile app is still valid.

In the event the timer expires, the user is prompted that access to the network is required to verify their version. If an update is available, the user will be asked to accept; if they decline, the mobile app will fail to function. The choice to use such a mechanism should be based on risk. If it is more hazardous to deprive the user of the use of the mobile app than it is to permit the continued use of the old version, avoiding this approach should be considered.

Depending on the nature of the data, there may be a need for archiving of that data. A lower risk solution would involve designing the mobile app designed to periodically upload any data residing on the device. Retrieving data from devices at the retirement stage is likely to require user participation, e.g., to configure the application settings to allow uploading.

Risks associated with users continuing to run the mobile app after retirement should be considered and understood, as this may potentially leave the regulated company open to regulatory action or to private litigation.

Table 6.4: Risks Encountered During the Retirement Phase

Issue Type	Hazard	Risk
Technical	Lack of effective mechanism to remove the app from the market	Discontinued or unsupported mobile medical apps available in market place
	Inadequate archival technology	Retrieval of data not possible throughout the retention period
Supplier	Failure to withdraw from distribution channel (e.g., app store)	Discontinued mobile medical apps available in market place
	Inadequate archival process	Retrieval of data not possible throughout the retention period
User Risks	Lack of effective mechanism to remove the mobile app from the mobile device	User continues to use retired mobile app with unknown consequences
	Inability to disable active users	User continues to use retired mobile app with unknown consequences
	Deprive dependent user the use of a mobile app if no replacement available	Possible detrimental impact on patient health
Regulatory and Legal	Lack of understanding of medical device regulations	Failure to provide adequate documentation for medical device market withdrawal leading to non-compliance with medical device regulations
	Lack of understanding of local laws	Mobile app is not withdrawn in accordance with local laws (e.g., non-compliance with licensing agreements)

7 Appendix 2 – Quality Risk Management Approach

The following QRM approach is based on that described in GAMP® 5 [8] and is aligned with the ISO 14971 [11] framework and requirements as described in Section 4 Quality Risk Management for Mobile Applications.

It is an iterative approach used throughout the life cycle from concept to retirement.

This Appendix uses the following key terms taken from GAMP® 5 [8].

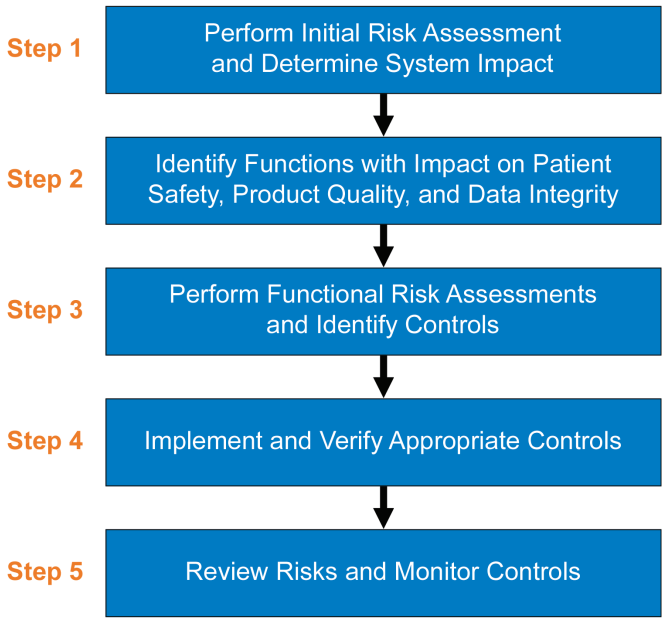
- **Harm:** damage to health, including the damage that can occur from loss of product quality or availability.
- **Hazard:** the potential source of harm (ISO/IEC Guide 51 [20]).
- **Risk:** the combination of the probability of occurrence of harm and the severity of that harm (ISO/IEC Guide 51[20]).
- **Severity:** a measure of the possible consequences of a hazard.

A five step process is suggested:

1. Perform initial risk assessment and determine impact
2. Identify functions with impact on patient safety, product quality, and data integrity
3. Perform functional risk assessments and identify controls
4. Implement and verify appropriate controls
5. Review risks and monitor controls

These steps are described in this appendix.

Figure 7.1: Quality Risk Management Process



7.1 Step 1 – Perform Initial Risk Assessment and Determine Impact

An initial risk assessment should be performed based on the defined and documented intended use. The initial risk assessment should be based on process risk information and product requirements.

The results of this initial risk assessment should include a decision on whether the application is GxP regulated and whether it is classified as a medical device. If the mobile app is a medical device, a determination should be made as soon as is possible in regard to which class of medical device it is. This can have a significant impact on documentation requirements and may include a requirement for premarket approval. Device classifications schemes vary depending on the applicable regulatory framework and the assessment should account for such variance depending on where a mobile medical app may be released. Medical device regulations vary from nation to nation; the requirements of each prospective market should be understood.

This initial risk assessment also should include an overall assessment of the level of impact on patient safety or public health, and also may consider other risk areas such as data privacy and security.

The level of effort, formality, and documentation of any subsequent steps should be determined based on level of risk and product impact.

If the nature of data to be collected by the mobile app is known at this point, a similar assessment of any potentially applicable data privacy rules is also essential. National data privacy laws vary significantly, and knowing these can be critical for requirements planning.

7.2 Step 2 – Identify Functions with Impact on Patient Safety, Product Quality, and Regulated Data Integrity

Product functions and features that have an impact on patient safety, product quality, and data integrity should be identified by building on information gathered during Step 1, referring to relevant specifications, and taking into account the product development approach, product architecture, and the nature of the hardware and software components involved.

This should be performed based on a substantially complete product specification.

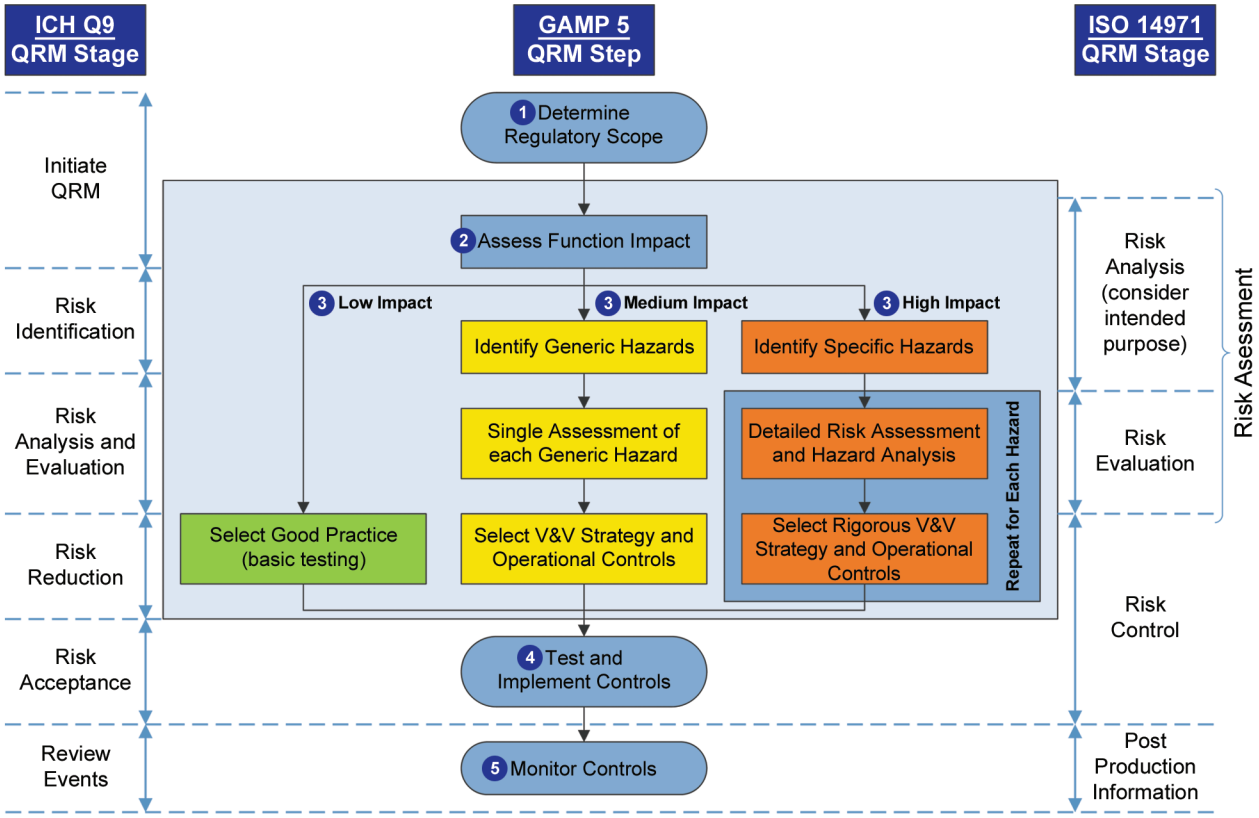
Any regulated electronic records and signatures should be identified. For example, if patient data related to a clinical study are being collected, the risk should be evaluated related to storing them on the device or transmitting them to a centralized database in real time. Both solutions have associated risks and benefits.

7.3 Step 3 – Perform Functional Risk Assessments and Identify Controls

Functions identified during Step 2 should be assessed by considering possible hazards, and how the potential harm arising from these hazards may be controlled.

It may be necessary to perform a more detailed assessment that analyzes further the severity of harm, likelihood of occurrence, and probability of detection (GAMP® 5 [8], Appendix M3 provides one example of such an approach).

Figure 7.2: Relationship between GAMP® 5, ISO 14971, and ICH Q9 Approaches



V&V = Verification and Validation

The decision as to whether to perform detailed assessment for specific functions should be dealt with on a case-by-case basis and documented.

Appropriate controls should be identified based on the assessment. A range of options is available to provide the required control depending on the identified risk. These include:

- Restriction of intended use
- Restriction to intended users (e.g., a medical app available only by prescription)
- Modification of application functionality and features
- Selection or rejection of specific platforms
- Modification of system design or architecture
- Limiting or controlling availability of the application
- Handshake protocols to verify that the combination of app/hardware/operating system is compatible and supported.
- Modification or enhancement of user instructions
- Increased rigor of testing

7.4 Step 4 – Implement and Verify Appropriate Controls

The control measures identified in Step 3 should be implemented and verified to ensure that they have been successfully implemented. Controls should be traceable to the relevant identified risks.

The verification activity – typically design review and design verification as well as testing - should demonstrate that the controls are effective in performing the required risk reduction.

7.5 Step 5 – Review Risks and Monitor Controls

Following testing and implementation of required controls, residual risk should be evaluated. If residual risk is not acceptable, additional controls may be required.

A process should be established to collect and review information about the product following release. Information about potential risks related to the mobile application in the market should be gathered and evaluated. Previously unanticipated hazards or examples of risk control failure should be reviewed, and appropriate new or modified risk control measures identified, if necessary.

8 Appendix 3 – Requirements Definition for Mobile Apps

8.1 Overview

Requirement gathering and definition activities typically start during the Concept Phase and are finalized during the Production Phase.

Detailed requirements for medical devices will vary from region to region, and it is the responsibility of organizations and individuals involved to identify, interpret, and apply such requirements as applicable. In all cases, companies should be familiar with the local regulations in any country where they intend to deploy a mobile app.

While new requirements may be identified at any time, most should be identified early in the development process. The product quality plan (required for mobile medical apps) should specify a point by which the identification of new requirements is substantially complete. This may lead to multiple endpoints if an iterative approach is used.

Developing product requirements should begin with an application definition statement.[5] This is a concise definition of a mobile app's main purpose and its intended users. The application definition may be part of the product requirements specification or separate.

Product requirements are developed by gathering and defining user needs, intended use, and regulatory requirements. The requirements should define clearly and precisely what the mobile app should do and state any constraints. Requirements should be written such that they can be tested. Each requirement should be assigned a unique identifier and should be traceable through specifications and testing throughout the entire life cycle of the application. Requirements should be reviewed and approved.

- Requirements may be developed internally by the supplier or may be provided by the regulated company and should involve the key stakeholders associated with the development process.

- Changes to requirements should be controlled. Changes to subsequent specification documents that affect the requirements should lead to an update of the requirements. The product requirements documentation should reflect the current version of the mobile app. Every software release should have a corresponding approved product requirements specification.

For mobile apps, usability and human factors should be addressed during requirements capture and definition. Appropriate user evaluation activities and techniques should be included. For example, usability factors could involve ensuring that all screen content is readable on various screen sized and resolutions, and that links and soft buttons are sufficiently far apart that selecting the desired option is straightforward.

(ISO/IEC 62366 [21] specifies a process for a manufacturer to analyze, specify, design, verify, and validate usability, as it relates to safety of a medical device).

Information obtained from initial risk assessment activities may be used to develop requirements. Requirements definition and risk management will be concurrent and interact, following an iterative process to refine and finalize.

[5] An application definition statement is a term introduced as part of Apple's® developer's toolkit. It is a concise, concrete declaration of an app's main purpose and its intended audience. Apple's® advice is: "Create an app definition statement early in your development effort to help you turn an idea and a list of features into a coherent product that people want to own. Throughout development, use the definition statement to decide if potential features and behaviors make sense."

For regulated mobile apps, regulatory/statutory requirements should be clearly identified as such. If the mobile app maintains any information that is specifically required by a predicate rule, this may bring it into scope of applicable electronic record and signature requirements, such as US FDA 21 CFR Part 11 [16] and EU GMP Annex 11 [18]. Requirements for any technical controls needed to enable compliance with these regulations should be defined. See GAMP Good Practice Guide: A Risk-Based Approach to Compliant Electronic Records and Signatures [19] for further details.

The product requirements specification should consider:

- Intended uses, including clinical objectives if applicable
- Brief functional description
- Key features and characteristics
- Functional and performance requirements
- Human factors/usability requirements
- Data requirements
- Interface requirements
- Security requirements
- Retirement or withdrawal requirements (e.g., a mechanism for disabling the mobile app).
- Other quality/non-functional requirements

Requirements should be prioritized (e.g., critical, important, and desired).

See GAMP® 5 [8] Appendix D1 for further details on user requirements specifications.

In general, the principles of specifying requirements for a mobile app are the same as for other software based systems or product, but some areas deserve special emphasis because of the nature of the product. These are discussed in this appendix.

8.2 Prototyping

Prototyping methods may be used to clarify user requirements or to evaluate areas of risk. Typically, a prototype is used to evaluate:

- The acceptability of a user interface
- The performance of critical algorithms
- Suitability of the overall solution
- Aspects of system performance, e.g., capacity and speed

The aims and objectives of the prototype should be clearly defined in order to be effective. The prototype should be evaluated against these to ensure that the objectives are met. Suppliers should define how information gained can be incorporated in a controlled manner into specifications for the final product. This requires rigorous version control and segregation of prototype and final software.

Making specific data or functionality available on a mobile device in a usable way can be challenging given the limited space for interaction and the expectations of the users on specific platforms.

Usability needs and requirements for mobile apps should be addressed by specific and defined prototyping activities aimed at identifying an appropriate interface and verifying the usability of the planned interface with the users.

This can be done in stages with increasing degrees of detail, but should include evaluation of an interface working on the mobile device with sample data. This is necessary because the interactive nature of an application means that the only way of confidently assessing the reaction of users to the proposed interface is with a working interface.

This prototyping and evaluation work should be performed after the main functionality of the mobile app has been identified and documented in an application definition statement and typically, in a set of use cases or scenarios. The prototype should cover all the identified use cases and should be evaluated on a typical cross-section of users.

There may be a series of prototypes with the prototypes being updated dependent on the evaluation of previous prototypes. Each evaluation can use a small set of users, as long as different users are evaluated for each prototype – 5 to 10 users is usually sufficient for each evaluation cycle.

Evaluation should take the user through the use cases for the mobile app, ensuring that:

- Users understand how to carry out key tasks and can perform them
- Available actions and feedback are clear to the user
- The typical user can read information as presented in the interface
- Interface items such as buttons, pick lists, gestures can be successfully performed by the user population

The main mobile platforms (iOS®, Android®, Windows®) have defined user interface guidelines with different user interface expectations; therefore, it is recommended to perform the evaluation separately for each platform supported although a degree of commonality and similarity may be expected for the same mobile application running on different mobile platforms.

The results of this prototyping and evaluation work should feed into the requirements for the mobile app.

At any point in the prototyping process, a decision may be made that the prototype has evolved to the point that it has to become a formal design. At this time the Design History File (DHF) should be established or updated.

8.3 User Interface Requirements

The prototyping and evaluation of the user interface performed to enhance usability should result in a more detailed documentation of the user expectations of the product. An effective way to achieve this is by generating a set of use cases (definitions of interactions between systems and users in a particular environment and related to a particular goal) with detailed sets of screens explaining what the product does for each use case.

The relevant manufacturer's interface and design guidelines for the platform(s) in question should be followed, e.g., Apple *iOS Human Interface Guidelines* [22], which describes the guidelines and principles of user interface for iOS apps; or the equivalent guidelines for Android® or Windows®.

Verification of user interface requirements can be achieved via human factors analysis, e.g., making sure that buttons and fields are designed to be a usable size.

8.4 Connectivity Requirements

If the mobile application needs to connect to the internet or another device to either access data or to upload data, the requirements process should identify those needs and also define what the mobile app does when the remote service is unavailable. For mobile medical apps, special attention should be given to the issue of patient impact if communications fail.

8.5 Data Management Requirements

Significant attention is required for the planning of data management controls and practices (see section 5.3 for data life cycle description).

8.6 Target Platform Requirements

- The versions of operating systems and which platforms should be targeted and supported should be carefully considered during product quality planning, due to the evolving mobile platforms environment. The operating systems and platforms should be clearly defined in product requirements.

- The situation is different between major platforms (e.g., the range of device attributes may be wider, including different screen dimensions, physical keyboards, features such as NFC, the presence or lack of features such as accelerometers, and overlaid features).

- It is recommended that a default strategy and requirement set is defined and maintained identifying constraints, characteristics, (e.g., “screen resolution at least 960 by 480”) and the current operating systems versions supported.

The battery demand for a mobile app may be greater than is typically required for the mobile device to function in its normal capacity. In such cases, a requirement should be noted that a battery warning is needed before the power capacity falls below the required minimum to run the mobile app.

8.7 Requirements for Application with Associated Device (Non Stand-Alone)

Where the development of the mobile app includes interfacing to additional equipment, extra considerations should be taken into account during the requirements stage, e.g.:

- Limiting the mobile devices that can drive the equipment (so other mobile apps or devices cannot work with the equipment)

- Limiting the equipment that the mobile app can drive (so that the application cannot be used with non-standard equipment)

- Need for calibration

This is likely to take the form of an encrypted handshaking process between the mobile app and the equipment.

8.8 Mobile Apps Retirement and Withdrawal

Requirements for the mobile app should consider mechanisms that allow for its retirement either forcibly or automatically, such as in the event the application is no longer supported.

8.9 Mobile Apps Corrective or Removal Actions

If the mobile app is classified as a medical device, the requirement process should identify the mechanism to remove the mobile app from the market or potentially to stop its use if the corrective action is not accepted.

8.10 Labelling Requirements

Any required medical device labelling information needs to be defined and provided with the mobile app. The user should be able to access the full label via mobile app function at any time.

9 Appendix 4 – Mobile App Architecture

There are several high-level architecture concepts to consider when planning a mobile app. These are a result of the different environments in which a mobile app may be used, such as different types of mobile devices and different networks.

The most important factors to consider are:

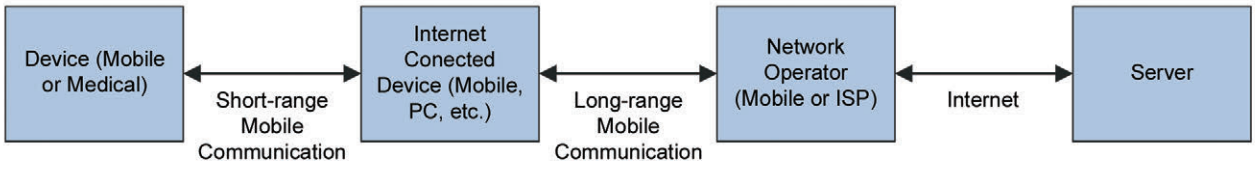
- Device connectivity
- Device components
- Mobile client approach

9.1 Device Connectivity

Not all mobile apps require communications, but it is common that the connectivity inherent in the platform will be leveraged. Mobile devices can communicate via modes that include both long and short range technologies. Long-range communications are via wireless networks (GSM, GPRS, Edge, 3G, 4G), or WLAN to an internet connection. Short-range communication protocols include Bluetooth, NFC, and potentially others. Usually the short range communications are between devices, e.g., between two mobile devices, mobile device/PC, mobile device/medical device. Where mobile apps employ short-range communications, data exchanged via short-range communication may be sent somewhere by the partnered device via one of the long-range communication protocols.

This traffic will go via a network operator to the public Internet, a VPN, or leased line to connect to the application server (see Figure 9.1). In Figure 9.1, the first device could be a mobile device or a medical instrument that is passing information via a short range communication protocol to a device that is connected to the internet. This second device could be a mobile device or a PC. This device transmits information (wirelessly or via wired connection) to a mobile network operator or ISP, which automatically passes the information to a target server.

Figure 9.1: High-Level Architecture of How a Mobile Device Reaches an Application Server or an Application Store



9.2 Device Components

There are a number of different components (or assets) that the mobile client can make use of that may differ between manufacturers and models, such as the hardware, operating system and the native browser. These differences should be considered when designing the mobile client.

- Important hardware considerations include screen size, memory, processor power, sensors such as GPS or motion sensors and radio modules (e.g., Wi-Fi, GSM, 3G, 4G, Bluetooth)
- Operating systems (e.g., Apple iOS®, Android®, Windows®) will differ between vendors and versions.
- Native browser capabilities also differ between browser make (Safari®, Chrome®, Firefox®, Explorer®) and versions

9.3 Client Approach

A mobile client can be designed to leverage varying levels of server-based back-end computing power (see Figure 9.2). A thin client (also referred to as a “mobile web app”) is essentially only a browser front end, configured specifically to address the needed functionality. All data processing and data storage occurs on a web server.

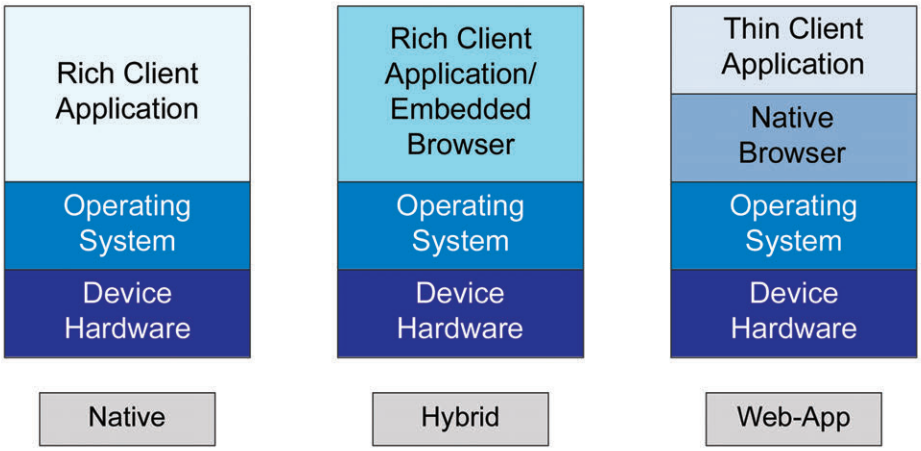
The rich client (also referred to as “thick” or “native”) will generally have the functionality embedded in the application residing on the device, and there will usually be locally stored data. There may be no web back end although it could involve cloud-based storage.

A hybrid approach involves a mobile client which could be designed as a native application with an embedded browser that can interact with a remote server from within the native application. Data processing and/or storage could be distributed between the mobile app and the server.

The complexity of the back-end can vary significantly based on this architecture choice. Assuming identical user requirements, the back end will generally be most complex for the thin client; somewhat less so for the hybrid; and least so for a rich client. However, any one of these architectures could be augmented by considerable server-based computing power where demanded by the user requirements.

Validation planning should account for the client architecture so that appropriate focus is placed on the highest impact areas of the complete computing environment.

Figure 9.2: Building Blocks for Rich and Thin Mobile Client Mobile Devices



Combining the connectivity options with the client options provides three distinct cases:

1. Unconnected rich client
2. Hybrid client
3. Connected thin client

9.3.1 Unconnected Rich Client

The mobile client conducts all processing locally and is not designed to interact with a server after being downloaded. It will be able to make use of most device assets because it runs on the operating system of the mobile device. This has both advantages and disadvantages:

Table 9.1: Advantages and Disadvantages of an Unconnected Rich Client

Pros	Cons
The mobile app will run regardless of connectivity being available or not.	The mobile app is difficult to maintain, update, recall and track.
Battery consumption may be improved without communication although this could be offset if significant local processing is required.	The client data will be lost if the device is changed, lost, stolen, or broken.
The client costs less to run as no mobile charges will be incurred.	The client may need to be adapted considerably between different mobile devices.
Richer user experience as client is customized for a specific mobile device or device family.	

9.3.2 Hybrid Client

The mobile client conducts some processing locally and it connects with a server, for instance, to a database or for further processing. Because it runs on the operating system, the mobile device will be able to make use of most device assets. The advantages of this architecture are similar to the unconnected rich client.

Table 9.2: Advantages and Disadvantages of a Hybrid Client

Pros	Cons
The mobile app could be designed to run regardless of the availability of connectivity.	Some processing or storage capability may be unavailable if not connected.
The mobile app may be easier to maintain, update, recall and track.	The client may use more energy as some communication is needed.
The user can have a richer experience as the mobile app is customized for a specific mobile device or device family.	The client will incur some mobile charges.
Locally retained data could allow a mobile app to continue to run if inadvertently disconnected from the network, and any collected data could be uploaded when next connected.	The client may need to be adapted considerably between different mobile devices.
The client data is backed up if the device is changed, lost, stolen, or broken.	Potential connectivity issues.

9.3.3 Connected Thin Client

A mobile thin client is more than a browser; this terminology implies that there is a mobile app that acts primarily as a dedicated portal, perhaps including a security layer, that connects to a web server where processing occurs. Generally, because it is run on the device browser, the mobile client will only be able to make use of device assets that the browser can access.

Table 9.3: Advantages and Disadvantages of a Connected Thin Client

Pros	Cons
The client is easy to maintain, update, recall, and track.	The client will not run without connectivity.
The client data is backed up if the device is changed, lost, stolen, or broken.	The client may use more energy as some communication is needed.
The client needs to be adapted less between different mobile devices.	The client will incur some mobile charges. Bandwidth availability may be an issue, leading to possible communication challenges.
	Different platforms may render screen views differently, resulting in potentially compromised data displays and functionality.

The approach to validation and maintaining a compliant state will be affected by the architecture. While rich clients will usually add complexity to the support model and therefore, to the maintenance of a validated state, the overall validation effort will primarily depend on the purpose of the mobile app. For thin client architectures, many of the validation requirements will be similar to standard application validation as described in GAMP® 5 [8].

10 Appendix 5 – Production Phase for Mobile Apps

10.1 Overview

The Production Phase involves the refinement and finalizing of requirements, supplier assessment and selection (if the product is not being developed internally by the regulated company), various levels of specification, design, coding, and verification leading to acceptance and release for operation. Supporting activities include QRM, traceability, and design review.

Product specification and design usually consists of an iterative series of activities[6] that starts when user needs are gathered and is completed when a mobile app is approved for release to the market.

Prototype designs may be created and evaluated until a final design is identified that meets established requirements. Requirements, specification, and design may be continuously assessed and refined (revised) until an appropriate, complete, and consistent design configuration is achieved.

See Appendix 3 for further information on requirements definition for mobile apps.

An established process for change management during product development should be applied to all specification and design deliverables (See GAMP® 5 [8] Appendix M8 for guidance).

The specification and design process should be integrated with the QRM process. The required controls identified during risk assessments should be captured in the appropriate documentation. Risks that are not sufficiently mitigated through product design should be noted and referenced for external mitigation (e.g., through procedural controls, mobile service provider controls). For further guidance on risk assessment for medical devices, see ISO 14971 [11].

For mobile medical apps, documented design reviews should be performed, and the relationships between product requirements, specification, design, risk assessment, and testing activities should be summarized, e.g., in a traceability matrix.

Before formal product testing activities begin, product specifications and designs should be reviewed and approved and the code placed under change control.

A prototyping approach can allow for testing before all design activities are complete, as long as the design for the module being tested has been approved. All specification documents should follow established document control and change management procedures. Specifications should be updated as necessary when existing requirements need modification, or when requirements need to be added or deleted.

The aims and objectives of the prototype should be clearly defined in order to be effective. The prototype should be evaluated against these to ensure that the objectives are met. How information gained can be incorporated in a controlled manner into specifications for the final product should be defined. This requires rigorous version control and segregation of prototype and final software.

Production phase activities may be shared between the supplier and the regulated company and should involve the key stakeholders associated with the development of the mobile app.

[6] Iterative approaches to mobile app development are generally more common than a linear waterfall approach. When used in the development of medical device apps; however, controlled updates to the DHF are required with a clearly defined linkage between development iterations and design versions. This should be defined as part of the life cycle and development approach.

10.2 Specifications

The mobile app specification and design should be described in terms that will allow demonstration of conformance to product requirements during product testing. Such documents typically include:

- Functional specifications
- Design specifications

Functional specifications should clearly and completely describe what the product will do. They should be produced such that objective testing can be subsequently performed.

Design specifications should be based on the functional specifications and should be sufficiently detailed so that the product can be developed.

Specifications should be reviewed and approved with traceability established between related documents. They should be managed under change control with the awareness that change to one document may lead to a change being required in others.

Specifications may be covered by one or more documents depending on complexity and risk. A prototyping strategy will result in evolving specifications. At the end of the development process; however, approved master copies of the specifications are an expectation.

It is recognized that not all suppliers use the specification terms used in this Guide, but may still meet the objective of providing adequate specifications through the provision of other documentation.

10.2.1 Functional Specifications

The functional specification should describe those requirements that will be implemented in the design and should be based on the product requirements specification. Functional specifications should clearly and completely describe what the product will do. They should be produced in a way that ensures objective testing can be subsequently performed.

For further guidance, see GAMP® 5 [8] Appendix D2.

10.2.2 Design Specifications

Design specifications should be based on the functional specifications and should be sufficiently detailed so that the product can be developed.

During the design process, specifications are translated into a logical representation of the software to be implemented. The design specification is a description of what the software should do and how it should do it. Design documentation should cover data flows and interfaces.

For further guidance, see GAMP® 5 [8] Appendix D3.

10.3 Design Reviews

Design reviews evaluate deliverables against standards and requirements, identify issues, and propose required corrective actions. They are planned and systematic reviews of specifications, design, and development, and should be planned to occur at suitable stages during the life cycle. While this is good practice for all mobile applications, it should be noted that design review is a formal requirement of the medical device regulations (e.g., 21 CFR Part 820.40(h) [13]).

Design reviews aim to identify and eliminate issues that would otherwise lead to changes at a later stage.

For further guidance, see GAMP® 5 [8] Appendix M5 and relevant regulations and standards, such as 21 CFR Part 820 [13] or ISO 13485 [10].

10.4 Traceability

Traceability establishes the relationship between two or more products of the development process. Traceability ensures that requirements can be traced to the appropriate specification and design elements, and that requirements can be traced to test activity that shows that requirement has been met.

Product requirements, functional specification, and design specifications are closely related and frequently referenced during the development process. A traceability matrix is a living document that represents the relationships between these components and to testing activities. Traceability may be recorded in a separate tool (e.g., as part of an integrated software development environment) or document, or it may be integrated into the specification and test documents. Traceability evidence should be maintained for each product release.

For further guidance, see GAMP® 5 [8] Appendix M5.

10.5 Software Production

The supplier should establish and maintain a formal system for controlling software production.

Appropriate methods and tools should be used and the use of these should be documented.

Rules and conventions, such as acceptable languages, coding standards, and naming conventions should be established. Code reviews or walkthroughs should be performed.

Tools recommended and/or provided by the mobile device platform manufacturer should be used. Such tools will typically include features that support source code control, configuration management, unit testing, emulation, and debugging on real devices.

For further guidance, see GAMP® 5 [8] Appendix D4.

10.6 Testing

Mobile apps should be tested against formally approved test plans and specifications. Testing should demonstrate that all product requirements, functionality, and design activities have been successfully met. Appropriate regression testing should be performed when changes are made.

The testing of mobile apps that are components of systems used to support GxP will typically be addressed as part of wider system verification or qualification activities as defined in the regulated company QMS, and the testing guidance in GAMP® 5 [8] should be followed.

For mobile medical apps, product testing should meet the requirements of design control by confirming that design output meets the design input requirements, that products conform to define user needs and intended uses, and that product design is correctly translated into production. For mobile medical apps, test records are product acceptance records under 21 CFR Part 820.70 [13] and subject to 21 CFR Part 11 [16].

Product testing entails running software products under known conditions with defined inputs and documenting outcomes so that they can be compared to their pre-defined expectations and acceptance criteria.

Product testing may involve one or more stages of testing, depending on the nature and complexity of the product. Typically, a mobile app will require at least unit testing and product testing.

Product testing should include either actual or simulated use of the software being tested within the context in which it is intended to function. Product testing should be conducted on a representative range of intended devices. This may add complexity for product testing when there are multiple platforms supported, and may involve third parties that specialize in testing on different platforms. Testing involving members of the target user community should be considered for final product testing.

Prior to conducting final product testing, the product requirements specification, functional specification, and design specification should be approved.

When selecting and planning the appropriate product tests, the range of potential patient populations, intended uses, and clinical/physician practices (if relevant) should be considered. If clinical trials involving human subjects are necessary, proper controls for clinical usage, investigation of failures, and any patient follow-up should be in place.

All implemented product requirements should be taken into consideration during product testing.

Test records should be reviewed and approved and retained for a defined period that meets company and regulatory record retention requirements (in compliance with local record retention schedules and not to be shorter than the lifetime of the software). Test results may be summarized in a test report or other report.

The user instructions should be evaluated during product testing to ensure that product features, step by step instructions, appropriate warnings and precautions, and other risk mitigations requiring labelling are present, readable, understandable and accurate.

Test incidents and failures should be reviewed and managed in accordance with a formal documented process that includes appropriate investigation; traceability to any defective specification(s), test scripts, or use cases; corrective actions including update of specifications, scripts or use cases; and retesting as appropriate.

Test tools and environments used to conduct testing must be formally defined and assessed for adequacy for the intended testing. Test tools and environments recommended by the relevant mobile platform manufacturer should be used where possible.

For further guidance, see GAMP® 5 [8] Appendix D5. For detailed information, see also the GAMP® Good Practice Guide: A Risk-Based Approach to Testing of GxP Systems (Second Edition) [23].

10.6.1 Stages of Testing

Depending on the complexity and risk of the mobile app, testing may consist of:

- Unit testing
- Integration testing (where necessary)
- Product testing

Testing a mobile app should include unit testing and product testing. For mobile apps that include an associated device or access web services, there also will be a requirement to perform integration testing. For most mobile apps, the small scale of the mobile app may mean that a separate test strategy or test plan document is not required.

Unit testing follows the same principles as for other types of systems and products and tests code at the lowest testable level or unit. Such testing is typically automated, and the main application development toolkits typically include unit testing tools. Where the mobile app targets more than one platform (e.g., iOS® and Android®), unit testing should be performed on each platform and supported operating system.

Integration testing, whether with an associated device or with web services, is likely to be implemented as a set of automated unit tests. The tests should explore:

- The correct functioning of the associated equipment
- The loss of communication with the outside service, e.g., due to signal loss, bandwidth availability
- Error messages and failures from the outside service

System/product testing can cover many characteristics depending on the type of mobile app. The following checklist gives an indication of the characteristics to consider, but is neither intended to be complete nor exhaustive:

- **Interface:** the unit testing facilities do not use the interface directly, and so do not cover aspects such as whether buttons are visible on the screen, so functional tests should be repeated using the actual screen interface. That will include tests of all on screen facilities (buttons, sliders, landscape versus portrait display, etc.), as well as multi-touch gestures, accelerometer, NFC, etc., where used.
- **Coverage of platforms:** is the application tested on all significant delivery platforms (different models, different screen sizes, different operating system versions)? A technical judgment should be made as to how many hardware platforms need interface testing. This is especially appropriate for Android® platforms, for which user interfaces may be modified by service suppliers. This may require testing on some older device/operating system combinations of any type.
- **Functionality:** does testing cover the different availability of functions on different devices. For example, if the mobile app uses the iPhone's gyroscopic function - what does it do with a device that only has an accelerometer? What should happen to input from a device with a physical keyboard? If connectivity via short range communications (e.g., Bluetooth or NFC) are important to device function this should be tested.
- **Connectivity:** where the application connects to the outside world, the functionality under different kinds of connection should be tested (wireless, 4G or 3G, poor connection, no connection), e.g.:
  - Where the application is loading a web page and that page is abandoned by switching to another screen of the application, testing should ensure that loading of that web page is abandoned at that point (unless the expected behavior is continued loading for later viewing).
  - If data transfer rates need to be high, testing should verify that moving from 4G to 3G or from 3G to lesser coverage has no harmful effect.
- **Localization:** where the application is used internationally, test that all languages are shown correctly. In particular, the different size of words in different languages can cause problems, and testing should demonstrate they have been handled.
- **Time/Date:** where times are shown, test they are shown correctly depending on the country where the user is situated. Details of expected behavior will vary - a clock-based application should adjust the time when the time zone is changed; a fixed timetable should leave times in their original format. Tests should make clear the expected behavior.
- **Memory warnings:** mobile devices have limited memory, and occasionally devices will give memory warnings. Correct behavior in those circumstances should be defined and tested.

- **Battery warnings:** if the minimum battery capacity required for proper function is higher than the default warning for the mobile device, the battery warning function should be tested.

- **Interruption:** where the user can move away from an application and then back, or where the device can interrupt an application, testing should explore whether such interruptions are handled correctly.

- **Accessibility:** on iOS especially, there is an expectation that the application will work with the in-built accessibility model. This should be tested.

- **Updating:** where refresh of on-device data happens (either on start-up, or at set intervals, or through pull-to-refresh), testing should check that it occurs correctly.

- **Download:** if there are to be controls forcing users to download the latest version, testing should verify this process.

- **Other functions:** there are less common functions of mobile device that need to be specially tested if they are present – mail sending, location services, push notifications.

- **Training:** training materials, including help files and instructional videos, should be verified as correct.

10.7 Commercial Release and Distribution

Commercial release of the product should be performed in accordance with a formal process that describes criteria for release, responsibilities, records to be retained, and items to be released, including software and documentation.

Release notes defining fixes, changes, and new features should be compiled for each release, including minor releases and patches. It is possible that there may be different versions of these release notes, e.g., those directed at consumers and those directed at support staff. Appropriate management of this should be defined in supplier documentation.

Completing the validation of a mobile medical app may not mean that the app can be immediately released. Medical device regulations may have further requirements before the mobile medical app can be distributed, and local regulatory differences may mean that releasing the mobile medical app to the public may not be possible in all regions at the same time.

Distribution channels such as the Apple® App store or equivalent will impose specific rules that may impact the release process. The regulated company also may need to monitor the distribution channels to ensure that only approved versions are available for download.

For some mobile medical apps, it may be necessary to trace all installations (for recall/field notice purposes for example). Whether this is necessary should be determined, along with the processes that will be used to facilitate this defined and implemented.

10.7.1 Software Distribution

The process for distribution (push versus pull) should be defined and robust.

There should be a method for forcing distribution of an update if medically necessary. This could involve a push process that updates applications in the background, a feature that automatically checks for updates when the application is opened, a combination of the preceding, etc.

Specifically for mobile medical devices, controls should be in place that prevent download in countries where the device is not approved.

If a device is explicitly not supported, users should receive such notice when they try to download the application, and the download should be prevented.

The regulated company should have a process to assure that only approved versions are available for download from the distribution channels (e.g., the app stores).

Software distribution processes should include considerations related to application retirement as discussed in Section 5.2.4.

10.7.2 User Documentation and Training

The typical expectation of the mobile app user is that necessary training and documentation will be available as part of the mobile app, whether the device is linked to the internet or not.

One strategy is to identify new installations of the software on a mobile device and to offer introductory training at that point, which may be via a video. Another strategy is to have tips appear that can be easily turned off. Where the user has turned down the offer of training or turned off the tips, it should be possible to turn them back on later.

Initial identification of the training needed should be performed during the prototyping and evaluation phase of the project. The identified needs should be verified during testing.

11 Appendix 6 – Supplier Management and Good Practice

11.1 Introduction

This appendix is intended to provide guidance on good practice for suppliers of mobile applications. It also may be used as the basis for assessment of such suppliers. It is applicable to any supplier of mobile applications or provider of services associated with mobile application development, maintenance, support, or related web-hosting.

The term supplier may refer to any company, function or project within a regulated company that develops mobile applications or to any third party supplier contracted to provide products or services associated with mobile applications.

11.2 Overview

General supplier good practice as described in GAMP® 5 [8] is appropriate to the development of mobile applications. Specific characteristics (including the need for high levels of usability, platform specific user interface expectations, limited screen space for display, rapidly changing features and tools, undependable connectivity, and varying device capability) mean that there are specific demands on both the process and the developers.

Areas where mobile application development requires specific practices in addition to general supplier good practice include:

- Supplier assessment and selection:
  - Suppliers of mobile apps or services should be evaluated according the same general criteria as described in GAMP® 5 [8]. Any considerations unique to mobile apps should be addressed accordingly.
- Prototyping and evaluation:
  - This is among the most common of approaches to mobile application development. It is done to specifically address usability issues and is a necessary part of mobile application development.
- Requirements:
  - Many requirements will feed from the prototyping and evaluation, but specific non-functional requirements also should be considered. Proper documentation of new or modified requirements is important for regulated mobile applications.
  - Requirements may need to be addressed on different platforms via different designs, but should retain substantial similarities from the user standpoint, regardless of platform.
- Testing:
  - The characteristics of the domain lead to specific testing aspects that require specific consideration, e.g., relating to connectivity, multiple platforms.
- User documentation, support, and maintenance:
  - For most mobile apps, it would be expected that documentation, support, and feedback mechanisms are integral to the product. Extended support may need to account for issues peculiar to the life science industry, e.g., ensuring channels for the timely re-routing of misdirected adverse event reports.

- Responsibility for compliance with GxP or other requirements lies with the regulated company, but external suppliers may have considerable involvement in the process, and the adoption of good practices or otherwise could have a significant impact on the quality and compliance (where relevant) of the final product.

- Procedures adopted may be those of the regulated company or from the external supplier.

- Suppliers that provide services associated with mobile application development, maintenance, support, or web-hosting should have an appropriate approach to ensuring adequate quality. For larger companies, this should entail a documented QMS. Small development companies (e.g., garage developers) may not have a documented QMS. It may be acceptable to use such companies, but the regulated company should ensure that the required quality management activities are considered adequately.

- **Note:** for mobile applications that are medical devices, the developing company is subject to inspection by regulatory authorities for compliance to device quality regulations. The status as manufacturer of a medical device cannot necessarily be transferred from the company that develops the software to a regulated company (e.g., 21 CFR Part 807.20(a)(1) and 21 CFR 807.20(a)(2) [24]).

11.3 Supplier Good Practice

Table 11.1 summarizes supplier good practice activities that apply to mobile app development and support.

Table 11.1: Supplier Good Practices

Practice	Description
Establish QMS	The supplier QMS should: • Provide a documented set of procedures and standards • Ensure activities are performed by suitably competent and trained staff • Provide evidence of compliance with the documented procedures and standards • Enable and promote continuous improvement
Establish Requirements	The supplier should ensure that clear product requirements are defined or provided.
Quality Planning	The supplier should define how their QMS will be implemented for a particular product, application, or service.
Assessments of Sub-Suppliers	Suppliers should formally assess their sub-suppliers as part of the process of selection and quality planning.
Produce Specifications	The supplier should specify the system to meet the defined requirements.
Perform Design Review	The design of the system should be formally reviewed against requirements, standards, and identified risks to ensure that the product will meet its intended use.
Software Production/Configuration	Software should be developed in accordance with defined standards, including appropriate code review processes. Appropriate tools shall be used to support software production activities.
Perform Testing	The supplier should test the system in accordance with approved test plans and test specifications.
Commercial Release of the System	System release to customers should be performed in accordance with a formal process.
Provide User Documentation	The supplier should provide adequate product documentation.
Support and Maintain the System in Operation	The supplier should support and maintain the system in accordance with established processes. The process for managing and documenting system changes should be fully described.
System Replacement and Retirement	The supplier should manage the replacement or withdrawal of products in accordance with a documented process and plan.

11.3.1 Supplier Quality Management System

Suppliers should follow an established quality approach. Consideration should be given to following a recognized standard, e.g., ISO 13845 [10] or an equivalent. The regulated company has the responsibility to determine whether formal certification against a standard is required; this is mandatory in some countries for particular classes of medical device. In general, the quality approach should define:

- The process being followed to deliver and support the product, application, or service
- Responsibilities, including clear separation of authority between quality assurance and other groups, such as product development, product support, finance or marketing
- Deliverables
- Documentation
- Planned reviews of the quality approach and internal audits
- Approach to continuous improvement of the quality approach and its use

Regulated companies who have elected to use a small development company, e.g., a garage developer, may need to accept some level of risk related to the quality management process, including taking on some of the quality management activities which support the development process.

The quality approach should be based on a life cycle concept for the development and subsequent support of the product. This Guide does not recommend any particular life cycle or development methodology, but rather highlights those activities expected of suppliers.

The quality approach should include formal procedures covering the activities supporting system development, including:

- Requirements, specification and testing
- Software management, control, and release, including all versions currently supported for all platforms and operating systems
- Distribution of the software
- Development change control
- Configuration management
- Traceability
- Training of supplier staff
- Document management
- Backup and restore

If a mobile app is part of a larger validated system, the supplier quality processes should be aligned with the quality processes of the regulated company responsible for the overall system.

11.3.2 Supplier Quality Planning

- Quality planning should define the activities, procedures, deliverables, and responsibilities for establishing delivery and monitoring of the service.

- The required information may be satisfactorily covered by other contractual documents, such as a Service Level Agreement (SLA), in which case a separate plan would not be required. SLAs should be established with all supporting organizations, as appropriate.

- The supplier should define how the QMS will be implemented for a particular product or service.

- This should include defining the life cycle model being followed and the project organization, activities, procedures, deliverables, and responsibilities for establishing the fitness for intended use.

- The approach for mobile application development should define software development techniques and tools. Prototyping can be a particularly effective tool for mobile application development.

- Quality management responsibilities (supplier versus regulated company) should be clearly defined.

- Quality requirements should be documented. In each case, the quality requirements should be clearly defined, reviewed, approved, accessible, and followed.

- See GAMP® 5 Appendix M6 for further details on quality and project planning.

11.3.3 Sub-Supplier Assessments

Suppliers should formally assess their sub-suppliers/sub-contractors as part of quality planning. Sub-suppliers/sub-contractors also should be periodically reassessed in accordance with the QMS. Regulated companies should not take on assessment of these sub-suppliers/sub-contractors; this is a supplier responsibility. However, the regulated company should ensure that suitable sub-suppliers/sub-contractors processes and quality standards exist and are followed.

The decision how to assess sub-suppliers/sub-contractors should be based on a documented risk assessment. If the regulated company wants any influence over the criteria for selection of sub-suppliers/sub-contractors, this should be stated clearly in the contract or signed quality agreement with the principal supplier.

The following aspects should be specifically considered if the sub-supplier/sub-contractor is being considered for mobile app development:

- How does the supplier approach the usability demands of mobile applications? An experienced supplier should have a credible strategy for how to produce a usable interface for a mobile app.

- What experience of tools has the supplier obtained from previous projects? The recommendation is that the native tools are used to build mobile apps for robustness – suppliers that already use and have experience of those tools have a significant advantage.

- What steps does the supplier take to ensure that staff is up to date with relevant technologies? The main native tools are evolving rapidly. A supplier of mobile apps should plan that staff keep up to date with the relevant tools through outside and internal training, and train staff to perform appropriate code reviews and walkthroughs.

- How successful has the supplier been in the past in producing useful and effective mobile apps? An overall assessment of advantages and drawbacks can be obtained from sources such as an app store (or equivalent) reviews. This means that careful exploration of previous submissions from the supplier can give a useful indication of the quality of functionality and interface that can be expected from that supplier.

Mobile medical apps sub-suppliers are expected to satisfy the same device GMP requirements as the primary manufacturer.

A common practice is to outsource testing. The contract with the principal supplier should be clear regarding expectations for testing. This includes who is responsible for developing test strategy, writing test scripts, and executing testing.

If the developer outsources **all** test responsibility, the organization responsible for testing should be considered a primary supplier on par with the developer. If the developer creates the strategy and scripts and contracts out test execution, the testing company can be a sub-contractor.

For further information on Supplier Assessments, see GAMP® 5 [8] Appendix M2.

12 Appendix 7 – Sample Mobile App Case Studies

This appendix contains hypothetical case studies that reflect a wide variety of types of regulated mobile apps. Some of them are medical devices, some of them are mobile medical apps, and some are regulated because they are components of computerized systems that have GxP or some other regulatory impact.

Note: these case studies are indicative examples only and many aspects of risk for the subject mobile app could vary significantly based on factors that include how the mobile app is used, who the users are, the nature of information processed or stored by the mobile app, what actions are taken based on such information, the architecture of the mobile app, and many other potential factors.

In some cases, the validation of the mobile app may lie with a party other than the regulated company for whom the mobile app was developed. Accountability for the mobile apps always lies with the regulated company. In the case of medical devices, the manufacturer (i.e., the mobile app developer) also bears responsibility for patient safety, data integrity, and documentation.

The case study examples provided are:

1. Post Treatment Therapy Aid
2. Remote Adverse Event (AE) Reporting
3. Integration with a Blood Glucose Meter
4. Direct Wireless Control of a Medical Device
5. Camera Used to Interpret Color on Test Strips
6. Medical Device Service Support
7. Man-Machine Interface (MMI) to a Manufacturing Control System
8. Dosage Calculator
9. Reporting for a Clinical Study
10. Sales Force Automation
11. Interface to the Warehouse Module of an ERP System

The case studies are detailed in table form in this Appendix.

Table 12.1: Mobile App Case Study Field Description

Mobile App:	Case Study Field Description
System Description:	A brief description of the mobile app. If there are several closely related uses or architectures, they will be noted here.
Potential Users:	People in whose hand the mobile app might be used
Medical Device:	Is the mobile app likely to be classified as a medical device? **Note:** this may not be an absolute; such classification can vary based on the way the app is used.
Medical Device Classification:	*This field is included in the case studies as a reminder that the regulated company should consider applicable local medical device classification during the Concept Phase. This is a critical point for determining a risk-based validation strategy, and it also has a large impact on the documentation required under device regulations. This field intentionally contains less detail in each case study to emphasize the dependency on both a very detailed knowledge of the device (far deeper than shown here) and the fact that various regulators have different standards and definitions.*
Electronic Health Record:	Classification as an EHR carries significant data privacy implications. Specific privacy requirements vary between regions.
Validation Required:	Validation is usually expected for GxP systems, including those with mobile components.
Validation Responsible:	The organization responsible for performing the validation. This is not always the same organization that is accountable for the validated state.
Calibration Required:	Some systems might require periodic calibration, e.g., a mobile app that reads color values from a chemical test strip.
Calibration Responsible:	The organization responsible for calibration. This is not always the same organization that is accountable for the calibrated state of equipment.
Production Phase Considerations:	Aspects that should be addressed during implementation. This could be related to any aspect of compliance and not just validation.
Operation Phase Considerations:	Aspects that need to be monitored and controlled during the production phase
Retirement Considerations:	Aspects that need to be addressed prior to and after retirement of a mobile app
Potentially Similar Systems:	Other mobile apps that may have similar uses or require similar controls. Not an exhaustive list.

Table 12.2: Mobile App Case Study – Post Treatment Therapy Aid

Mobile App:	**Post Treatment Therapy Aid**
System Description:	Suggesting, logging, and analyzing medical therapy or exercise regimes, based on user profiles. Variation: 1. Calendar-based therapy regimes (physical or medical) 2. Physical therapy plan modified based on feedback (e.g., heart rate) 3. Medication plan modified based on feedback
Potential Users:	Patient; healthcare provider
Medical Device:	1. No 2. Maybe (yes if the device replaces a human decision maker or if the device measures the parameter, e.g., heart rate), no if it provides information only) 3. Yes
Medical Device Classification:	To be determined during Concept Phase, if required
Electronic Health Record:	Yes if records are saved and/or transmitted to a health care provider (HCP)
Validation Required:	Yes
Validation Responsible:	Device supplier and user company
Calibration Required:	No
Calibration Responsible:	Not applicable.
Production Phase Considerations:	General fitness for intended use should be verified. Ensuring that recommendations reflect defined medical source reference material, and will not lead to injury. More rigorous validation for variation 2, greatest rigor for variation 3. Communications should be challenged.
Operation Phase Considerations:	Mechanism for software updates should be established. Change management should include management of mobile app through operating system upgrades. Consider how many versions of software need to be supported based on versions of mobile device operating system that are likely to be in the environment.
Retirement Considerations:	Local laws may affect the manner in which residual personally identifiable data on the device should be handled.
Potentially Similar Systems:	The systems noted below may be similar in nature, but the manner of use and the risk of the disease state will be relevant to validation strategy. • Mobile apps that perform simple calculations routinely used in clinical practice • Mobile apps that provide patients with simple tools to organize and track their health information • Mobile apps that provide or facilitate supplemental clinical care, by coaching or prompting, to help patients manage their health in their daily environment. See also the case study for dosage calculator.

Table 12.3: Mobile App Case Study – Remote Adverse Event (AE) Reporting

Mobile App:	**Remote Adverse Event (AE) Reporting**
System Description:	Supports drug or device safety processes, by enabling users to record and organize adverse event data, and use that data in the creation of submission reports required by regulators worldwide. Usually a global system located at one central site and accessed from many remote locations. Possible variations include: 1. Access via mobile web 2. Access via regulated company's mobile app 3. Access via third-party application that uses look-up function to associate a drug or device with the appropriate reporting channel
Potential Users:	Regulated company users; health care providers. Some systems may allow patients, families, and consumer advocates to report events.
Medical Device:	No
Medical Device Classification:	Not Applicable
Electronic Health Record:	No
Validation Required:	Yes
Validation Responsible:	Regulated Company
Calibration Required:	No
Calibration Responsible:	Not applicable
Production Phase Considerations:	Rigorous validation required. System maintains high impact regulated electronic records. Maintains patient data. System should be validated holistically, considering the mobile and non-mobile modules of the system together. Appropriate data integrity controls required. Validation considerations more complex for variation (3). Data integrity controls, e.g., acknowledgement at the mobile device end that a report has been received and passed technical quality checks, should be challenged.
Operation Phase Considerations:	If there are risks associated with using older versions of the mobile app, a mechanism should be devised to force upgrade or disable input from incompatible versions. Notification of the failure should include instructions on how to report the AE via other channels (e.g., phone, fax). Mechanism for software updates should be established. Change management should include management of mobile app through operating system upgrades. Consider how many versions of software need to be supported based on versions of mobile device operating system that are likely to be in the environment.
Retirement Considerations:	Not applicable. Records maintained on the central system.
Potentially Similar Systems:	Any system used in post market surveillance, e.g., GMP customer complaints

Table 12.4: Integration with a Blood Glucose Meter

Mobile App:	**Integration with a Blood Glucose Meter**
System Description:	Mobile app that can connect via Bluetooth or similar technology to a blood glucose meter. Data can be downloaded to a database on the device. Data can be sent from the mobile device to an HCP.
Potential Users:	Patient, HCP
Medical Device:	Yes
Medical Device Classification:	To be determined during Concept Phase
Electronic Health Record:	Yes if recorded and retained
Validation Required:	Yes
Validation Responsible:	Mobile app provider executes, regulated company reviews and approves
Calibration Required:	No
Calibration Responsible:	Not applicable
Production Phase Considerations:	System development processes need to comply with the appropriate medical device Quality System Regulation (QSR). Since this device could have possible life threatening consequences for specific failure modes (based on medical decisions based on bad data), validation strategy should be risk-based, thorough, and rigorous in those aspects that affect high risk functionality. Validation should concentrate on these primary aspects: • Demonstration that blood glucose readings are properly transmitted to and stored in the mobile device with the correct time stamp. • Demonstration that the Bluetooth connection is reliable, resistant to interference, and that incomplete data transmission would not lead to data corruption. • Demonstration that the mobile device database is adequately secure, including the inability of the device user to inadvertently delete the data via device configuration tools. Rules related to EHR require encryption, which should be challenged during validation. • Demonstration that data transmission to the HCP is secure and complete, and that if transmission is interrupted the situation is flagged or otherwise remediated.
Operation Phase Considerations:	Patient data is subject to privacy law, which may vary from nation to nation. Protections need to meet all local requirements. Any EHR requirements (e.g., for readability in various EHR formats) also need to be met. If there are risks associated with using older versions of the mobile app a mechanism should be devised to force upgrade or disable input from incompatible versions. Mechanism for software updates should be established. Change management should include management of mobile app through operating system upgrades. Consider how many versions of software need to be supported based on versions of mobile device operating system that are likely to be in the environment.
Retirement Considerations:	Archival of data directly from the device is probably not necessary; this should be managed within the relevant EHR system.
Potentially Similar Systems	• Mobile device recording vital signs • Blood pressure monitor interfaced to a mobile device • Mobile device receiving output from a heart monitor

Table 12.5: Mobile App Case Study – Direct Wireless Control of a Medical Device

Mobile App:	**Direct Wireless Control of a Medical Device**
System Description:	Clinician uses a mobile app to wirelessly adjust delivery of medicine via a syringe pump; clinician only supplies an output from the device to the syringe pump based on other readings/values seen on other pieces of equipment. This mobile app does not calculate dosages, only controls delivery.
Potential Users:	Clinician; health care professional
Medical Device:	Yes
Medical Device Classification:	To be determined during Concept Phase
Electronic Health Record:	No, unless a patient-specific history file is maintained
Validation Required:	Yes
Validation Responsible:	Developer/owning company
Calibration Required:	No (although the syringe pump would require calibration)
Calibration Responsible:	Not applicable.
Production Phase Considerations:	Device regulations apply and design verification activities are required. Simple one-way control of the syringe pumps is validated. Possible challenges could include different brands/ models of pumps.
Operation Phase Considerations:	Once released all changes should follow standard update procedures. • Change management should be in force • Versions of the app that should be supported based on; the applicability of the subsequent changes, and the versions of the devices in use. • Complaint monitoring should be in place. Mechanism for software updates should be established. Change management should include management of mobile app through operating system upgrades. Consider how many versions of software need to be supported based on versions of mobile device operating system that are likely to be in the environment.
Retirement Considerations:	Archival of data should be minimal as the mobile app only provides for an output. Records of treatment are likely to be in other applications with a possible exception of a patient-specific history as noted above, in which case EHR rules apply.
Potentially Similar Systems:	1. The above system postulates one-way communication. Two-way communication based on feedback (e.g., insulin delivery based in blood glucose readings from an interfaced device) adds significant complication and patient risk that must be addressed in validation. This also would likely affect the device classification level. It also could affect the likelihood that EHR comes into play, as well as the complexity of the records. 2. Device that reads blood pressure and sends to HCP for interpretation and dosage modification. In this case, unlike similar system #1, the dosage is not calculated and depends on physician feedback, so risk is lower.

Table 12.6: Mobile App Case Study – Camera Used to Interpret Color on Test Strips

Mobile App:	**Camera Used to Interpret Color on Test Strips**
System Description:	Mobile device camera used to interpret color of indicator strips (real-life urine analyzer)
Potential Users:	General public/patient that is under the supervision of a Primary Care Physician (PCP)
Medical Device:	Yes
Medical Device Classification:	To be determined during Concept Phase
Electronic Health Record:	No, unless a history is retained
Validation Required:	Yes
Validation Responsible:	Developer/owning company
Calibration Required:	Yes – camera (For this specific mobile app)
Calibration Responsible:	Mobile app owning company and user
Production Phase Considerations:	Device regulations apply and thus design verification activities are required. The mobile app is performing a calculation based on input it gathers from its attached input devices. Controls need to be defined and validated for the calibration procedure which could include such factors as light level, color interpretation, etc. A risk-based decision should be taken during development as to how often calibration is required, and as to what the mobile app will do if calibration is overdue.
Operation Phase Considerations:	Once released all changes should follow standard update procedures. • Change management should be in force • Versions of the mobile app that should be supported would be based on the applicability of the subsequent changes. • Complaint handling process Mechanism for software updates should be established. Change management should include management of mobile app through operating system upgrades. Consider how many versions of software need to be supported based on versions of mobile device operating system that are likely to be in the environment.
Retirement Considerations:	Archival of data, expiration of old versions.
Potentially Similar Systems:	If this functionality were to be expanded to function as other devices such as: hearing test, electronic stethoscope, sphygmomanometer, or a device suggests diagnosis based on picture of a skin lesion; then each of those functionalities would need to be calibrated on a routine basis and verified as correct.

Table 12.7: Mobile App Case Study – Medical Device Service Support

Mobile App:	**Medical Device Service Support**
System Description:	Mobile app supports field service engineers who support/repair/calibrate medical devices. Can be used to manage Field Corrective Action (FCA) documentation. Scenarios: 1. A standard browser provides access to a centralized system. All data storage is on the central system. This includes work orders sent to the technician as well as reports created by the technician. Inputs to the device include manually entered data plus barcode recognition using the device camera. 2. All information is stored locally on the mobile device until later upload.
Potential Users:	Service technicians
Medical Device:	No
Medical Device Classification:	Not applicable
Electronic Health Record:	No
Validation Required:	Yes
Validation Responsible:	Company providing service
Calibration Required:	No
Calibration Responsible:	Not applicable
Production Phase Considerations:	Calibration and FCA records are subject to regulatory inspection. Data integrity is a prime concern. 1. For mobile apps that are simply an interface to a centralized database, validation of the server-based components should follow normal GAMP® 5 [8] processes. Validation at the mobile device will concentrate on the capture and transmission of data. 2. For mobile apps that store information on the device, validation of the mobile app will focus on the integrity and functionality of the database, data capture, and transmission. 21 CFR Part 11 [16] audit trails will apply if the data is editable (it may be preferable to restrict edit capability until after the data is uploaded, to avoid supporting audit trails on the mobile device). If either scenario employs other than manual input (e.g., uses the device camera as a barcode reader to collect information on the equipment being serviced or the device uses wireless technology to download information), validation also should challenge that functionality and integration as well.
Operation Phase Considerations:	FCA, Preventive Maintenance (PM), and calibration records are controls for demonstrating compliance. Integrity of these processes and related data should be controlled. Validation periodic review should consider the system holistically and ensure that all aspects are under control.
Retirement Considerations:	Records are required to be retained for a defined retention period. When a mobile device is replaced, the process should include ensuring that any relevant records on the device are uploaded.
Potentially Similar Systems:	• Support app for field engineers for Good Manufacturing Practice (GMP) manufacturing or GMP/Good Laboratory Practice (GLP)/medical laboratory equipment • Calibration support tools that might be used by internal or external staff at a manufacturer

Table 12.8: Mobile App Case Study – Man-Machine Interface (MMI) to a Manufacturing Control System

Mobile App:	**Man-Machine Interface (MMI) to a Manufacturing Control System**
System Description:	Mobile app that provides an interface for an operator with a GMP manufacturing control system. Possible scenario variations include: 1. Access via web browser with view only rights 2. Access via web browser with edit rights for control settings 3. Native app with view only rights 4. Native app with edit rights for control settings
Potential Users:	Manufacturing operator or supervisor
Medical Device:	No
Medical Device Classification:	Not applicable
Electronic Health Record:	No
Validation Required:	Yes
Validation Responsible:	User company
Calibration Required:	No
Calibration Responsible:	Not applicable
Production Phase Considerations:	Validation strategy will vary with the scenario: 1. While there is no ability to affect the process directly, it is highly likely that decisions for process control will be based on system output; therefore validation concentrates on the website. Validation relevant to the mobile device concentrates on verification that the mobile browser displays the information properly, that it updates as necessary, and that flags notify a user when communications are interrupted. 2. As the only mobile app involved is the standard browser, as above validation concentrates on the server-residing software that actually controls the equipment and on the website through which interaction occurs. Testing from the device will be limited to verifying that information displays correctly and that two-way communications function effectively. 3. Same as 1, except that the subject of the validation is the mobile app rather than the website. 4. Full validation of the mobile app in the same manner as would be done for a server or PC-based control system, with the added aspect of the wireless communication
Operation Phase Considerations:	This is similar to any other wireless control system MMI. Ensuring real-time data is correctly displayed is critical because process adjustments will be made based on the information displayed via the mobile device. There must be adequate confidence that communications failures will not cause process problems. Security is also critical, as only authorized individuals can be allowed to adjust process parameters. There may be considerations relative to proximity: is it appropriate for an individual not at the manufacturing site to be able to adjust process parameters.
Retirement Considerations:	Archival of data not necessary; process data would reside in other systems.
Potentially Similar Systems:	• Interfaces to GMP or GLP laboratory control systems. • Interfaces to warehouse systems also would be treated similarly. • **Note:** interfaces with clinical laboratories would require additional privacy protections.

Table 12.9: Mobile App Case Study – Dosage Calculator

Mobile App:	**Dosage Calculator**
System Description:	Mobile app for the calculating of medication dosage. Three related, but different cases: 1. Based on patient gender and weight 2. Based on diagnostic information (e.g., Blood Pressure (BP) or blood glucose levels; used by the HCP) 3. Based on diagnostic information (e.g., BP or blood glucose levels; used by the patient)
Potential Users:	Patients and physicians
Medical Device:	1. No 2. Yes 3. Yes
Medical Device Classification:	To be determined during Concept Phase, if required
Electronic Health Record:	Only if the mobile app retains a log of dosages. If the mobile app sends such log information to a server, an EHR will reside on the server.
Validation Required:	Yes
Validation Responsible:	Supplier
Calibration Required:	No
Calibration Responsible:	Not applicable
Production Phase Considerations:	1. Validation is a very simple verification of a lookup table or calculation based on physical variables. 2. Validation should verify that the calculations properly account for all physical and disease state variables. The dosage calculation algorithm is likely to be significantly more complex than for case 1. In addition, if the physician can determine and set dosages remotely, verification of the communication channel is critical. Error conditions need to be fully understood and the mobile app must be proven to handle such conditions appropriately. 3. Similar to case 2 except that communication is not a factor.
Operation Phase Considerations:	For cases 1 and 3, if dosage recommendations are changed by the drug manufacturer or medical authorities, timely updating of the dosage algorithm will be necessary. For case 2, it can be assumed that this is a physician responsibility. Mechanism for software updates should be established. Change management should include management of app through operating system upgrades. Consider how many versions of software need to be supported based on versions of mobile device operating system that are likely to be in the environment.
Retirement Considerations:	Probably not critical unless dosage calculation algorithms need to be retired.
Potentially Similar Systems:	Mobile app interfaces directly with a medical device, either to receive diagnostic information (e.g., a blood glucose meter) or to send dosage parameter to a delivery system (e.g., a syringe pump). Such systems would introduce additional patient risk, and errors in the calculation or communication could result in under- or overdose.

Table 12.10: Mobile App Case Study – Reporting for a Clinical Study

Mobile App:	**Reporting for a Clinical Study**
System Description:	The mobile app is a platform from which a patient can report information related to a clinical study. Two basic scenarios: 1. Via a thin or rich client the patient enters information that is immediately sent to a server where the records are maintained and/or further processed. 2. Patient enters data to a rich client, which stores the information for later upload.
Potential Users:	Patients and clinicians
Medical Device:	No
Medical Device Classification:	Not applicable
Electronic Health Record:	1. No 2. Yes
Validation Required:	Yes
Validation Responsible:	Supplier and/or regulated company
Calibration Required:	No
Calibration Responsible:	Not applicable
Production Phase Considerations:	1. For a thin client, validation verifies that the mobile web browser interfaces properly with the centralized server. For a rich client, validation must account for differences in platform screens (are all data fields visible for all supported platforms) and that the mapping from the mobile app to the database server is correct. For all clients, possible communication error conditions should be challenged. If the system is intended to provide patient prompts, e.g., a reminder to take a dose, then that functionality should be challenged as well. 2. Roughly the same as for rich clients in case 1, plus privacy concerns related to EHR while the information resides on the mobile device.
Operation Phase Considerations:	If patients are involved in a long trial, there may be some who want to change mobile devices. This could introduce complications related to data migration. Privacy concerns if EHR is involved may require encryption of data. Mechanism for software updates should be established. Change management should include management of mobile app through operating system upgrades. Consider how many versions of software need to be supported based on versions of mobile device operating system that are likely to be in the environment. For rich clients that collect information that could comprise an adverse event report, data management concerns for that record need to meet AE reporting expectations, including requirements related to managing the mobile device-derived raw data of the AE report.
Retirement Considerations:	As each patient's mobile app instance would be keyed to a specific clinical trial, there is probably no harm in leaving the mobile app on patient's devices. This could be complicated somewhat if EHR is involved although the protections in place during the study should remain effective.
Potentially Similar Systems:	Not applicable

Table 12.11: Mobile App Case Study – Sales Force Automation

Mobile App:	**Sales Force Automation**
System Description:	A mobile device interfaced with a backend Customer Relationship Management (CRM) application is used by sales representatives as a sales aid. The device will be enabled to show presentations and videos, and also will be the tool through which a sales representative will enable physicians to order free samples of a medical product.
Potential Users:	Field sales force, customers (physicians)
Medical Device:	No
Medical Device Classification:	Not applicable
Electronic Health Record:	No
Validation Required:	Yes
Validation Responsible:	Regulated healthcare company, possibly in partnership with app supplier and/or supplier of the CRM software
Calibration Required:	No
Calibration Responsible:	Not applicable
Production Phase Considerations:	Functional specifications should clearly define what regulated activities (such as sample ordering or sample inventory management) are to be executed on the mobile device. Architecture will drive the validation direction, e.g.: • Will physician signatures related to ordering be captured as digitized physical signatures or via some other technology)? • How much processing is done on the device vs. the backend CRM application? Validation testing will be focused on the aspects of the mobile app with GxP impact only. However, the design of all system management controls over the mobile app and back-end software will need to comply with GxP standards
Operation Phase Considerations:	Change management processes need to ensure that appropriate regression testing is done whenever a change to a non-regulated portion of the system could impact the regulated piece. Sales representatives need to receive training in the use of mobile apps, including topics as varied as procedures for incident reporting and the need to accept software updates when they are pushed out to their devices. There may be data privacy concerns related to information stored on mobile devices related to Health Care Providers (HCPs). While this is probably not sensitive personal information, familiarity with the local laws of any region where the mobile app is used is necessary to assure compliance with those laws.
Retirement Considerations:	It is likely that all GxP data will reside in a back-end database so this is probably not an issue when a device is retired. However, compliance with all relevant data privacy laws could require that the data is wiped before disposal or re-purposing of the device.
Potentially Similar Systems:	Not applicable

Table 12.12: Mobile App Case Study – Interface to the Warehouse Module of an ERP System

Mobile App:	**Interface to the Warehouse Module of an ERP System**
System Description:	Through a tablet or smartphone a user can: • Scan barcodes or QR codes • Query status of warehouse inventory • Change status of inventory items • Record the results of visual inventory checks and adjust quantities • Initiate moves between inventory and staging areas • Confirm receipt of incoming shipments • Confirm departure of outgoing shipments
Potential Users:	Warehouse operators and supervisors, manufacturing planners
Medical Device:	No
Medical Device Classification:	Not applicable
Electronic Health Record:	No
Validation Required:	Yes
Validation Responsible:	User company
Calibration Required:	No
Calibration Responsible:	Not applicable
Production Phase Considerations:	Validation strategy will depend to an extent on whether this is architected as a mobile web app or as a custom designed interface. The latter may be somewhat more complex although that may be mitigated by the fact that the mobile app could be designed to run on only one or two mobile platforms. Because changing the status of inventory items (e.g., from quarantine to released) can have significant patient safety implications, not all staff should have that ability. Design and testing of security levels would be a critical concern. Barcode or QR code scanning also would be an important issue for validation, as misidentification of inventory lots could lead to incorrect dispensation of products. If a third party barcode/QR code scanning app is used the integration with the ERP system would be a point of challenge. If the architecture is not mobile web, the interfaces between the mobile app and the ERP system also would be critical.
Operation Phase Considerations:	Security management would be a principle concern, because the ability to change lot status must be restricted to those with authority to do so. While this would probably be managed within the ERP system, if there are any device-resident elements of the security controls, these need to be managed Unless a company operates within a BYOD framework, it is likely that the company will have control over the end-user device and can upgrade as needed. Careful management within a BYOD strategy would be required so the company may wish to make an exception and provide the device for this mobile app. The mechanism for software updates should be established. Change management should include management of mobile app through operating system upgrades. Consider how many versions of software need to be supported based on versions of mobile device operating system that are likely to be in the environment.
Retirement Considerations:	If the company controls the device, retirement should be easily managed via remote removal from user devices. As all relevant GMP data will reside in the ERP app, there is no archival issue.
Potentially Similar Systems:	• A mobile interface to a LIMS to support remote sampling and testing • A mobile interface to a building automation system

13 Appendix 8 – References

1. Federal Food, Drug, and Cosmetic (FD&C) Act, U.S. Food and Drug Administration (FDA), www.fda.gov.

2. Public Health Service (PHS) Act, U.S. Food and Drug Administration (FDA), www.fda.gov.

3. Mobile Medical Applications: Guidance for Industry and Food and Drug Administration Staff, 25 September 2013, U.S. Food and Drug Administration (FDA), www.fda.gov.

4. Guidance on Medical Device Stand-Alone Software (including apps), 19 March 2014, UK Medicines and Healthcare products Regulatory Agency (MHRA), www.mhra.gov.uk.

5. Guidelines on the Qualification and Classification of Stand Alone Software Used in Healthcare within the Regulatory Framework of Medical Devices, January 2012, European Commission DG Health and Consumer, http://ec.europa.eu.

6. Regulation of Medical Software and Mobile Medical "Apps," 13 September 2013 Australian Therapeutic Goods Administration (TGA), www.tga.gov.au.

7. Medical Information Systems – Guidance for Qualification and Classification of Standalone Software with a Medical Purpose, 31 January 2013, Swedish Medical Products Agency, http://www.lakemedelsverket.se/english.

8. *ISPE GAMP® 5: A Risk-Based Approach to Compliant GxP Computerized Systems,* International Society for Pharmaceutical Engineering (ISPE), Fifth Edition, February 2008, www.ispe.org.

9. ISO/IEC 62304:2006 Medical Device Software – Software Life Cycle Processes, International Standards Organization (ISO), www.iso.org.

10. ISO 13485:2003 Medical Devices – Quality Management Systems – Requirements for Regulatory Purposes, International Standards Organization (ISO), www.iso.org.

11. ISO 14971:2007 Medical Devices – Application of Risk Management to Medical Devices, International Standards Organization (ISO), www.iso.org.

12. 21 CFR Part 807 – Establishment Registration and Device Listing For Manufacturers and Initial Importers of Devices, US Code of Federal Regulations, U.S. Food and Drug Administration (FDA), www.fda.gov.

13. 21 CFR Part 820 – Quality System Regulation, US Code of Federal Regulations, U.S. Food and Drug Administration (FDA), www.fda.gov.

14. Australian Therapeutic Goods (Medical Devices) Regulations 2002, Section 4.4, Australian Therapeutic Goods Administration (TGA), www.tga.gov.au.

15. NIST Special Publication (SP) 800-124 Revision 1 – Guidelines for Managing the Security of Mobile Devices in the Enterprise, June 2013, National Institute of Standards and Technology (NIST), http://csrc.nist.gov/publications/PubsSPs.html.

16. 21 CFR Part 11 – Electronic Records; Electronic Signatures, US Code of Federal Regulations, U.S. Food and Drug Administration (FDA), www.fda.gov.

17. EudraLex Volume 4 – Guidelines for Good Manufacturing Practice for Medicinal Products for Human and Veterinary Use, Chapter 4: Documentation, January 2011, ec.europa.eu.

18. EudraLex Volume 4 – Guidelines for Good Manufacturing Practices for Medicinal Products for Human and Veterinary Use, Annex 11 – Computerized Systems, June 2011, ec.europa.eu.

19. *ISPE GAMP® Good Practice Guide: A Risk-Based Approach to Compliant Electronic Records and Signatures,* International Society for Pharmaceutical Engineering (ISPE), First Edition, February 2005, www.ispe.org.

20. ISO/IEC Guide 51:2014 Safety Aspects – Guidelines for Their Inclusion in Standard, International Standards Organization (ISO), www.iso.org.

21. ISO/IEC 62366:2007 Medical Devices – Application of Usability Engineering to Medical Devices, International Standards Organization (ISO), www.iso.org.

22. Apple iOS Human Interface Guidelines, https://itunes.apple.com/us/book/ios-human-interface-guidelines/id877942287?mt=11.

23. *ISPE GAMP® Good Practice Guide: A Risk-Based Approach to Testing of GxP Systems,* International Society for Pharmaceutical Engineering (ISPE), Second Edition, December 2012, www.ispe.org.

14 Appendix 9 – Glossary

14.1 Acronyms and Abbreviations

AE	Adverse Event
BP	Blood Pressure
BYOD	Bring Your Own Device
CDMA	Code Division Multiple Access
CPU	Central Processing Unit
CRM	Customer Relationship Management
DHF	Design History File
EHR	Electronic Health Record
ERP	Enterprise Resource Planning
FCA	Field Corrective Action
GCP	Good Clinical Practice
GLP	Good Laboratory Practice
GMP	Good Manufacturing Practice
GPRS	General Packet Radio Service
GSM	Global System for Mobile communications (originally Groupe Spécial Mobile)
HCP	Health Care Provider
HIPAA	Health Insurance Portability and Accountability Act (US)
LTE	Long-Term Evolution
MMI	Man-Machine Interface
NFC	Near Field Communication
OEM	Original Equipment Manufacturer
PC	Personal Computer
PCP	Primary Care Physician
PDMA	Prescription Drug Marketing Act (US)

PII	Personally Identifiable Information
PM	Preventive Maintenance
QA	Quality Assurance
QMS	Quality Management System
QR	Quick Response
QRM	Quality Risk Management
QSR	Quality System Regulation
RPO	Recovery Point Objective
RTO	Recovery Time Objective
SLA	Service Level Agreement
SME	Subject Matter Expert
SMS	Short Message Service
WLAN	Wireless Local Area Network

14.2 Definitions

Design History File (DHF) (21 CFR Part 820.30(j))

Each [medical device] manufacturer shall establish and maintain a DHF for each type of device. The DHF shall contain or reference the records necessary to demonstrate that the design was developed in accordance with the approved design plan and the requirements of this part.

GxP Regulation

The underlying international pharmaceutical requirements, such as those set forth in the US FD&C Act, US PHS Act, FDA regulations, EU Directives, Japanese regulations, or other applicable national legislation or regulations under which a company operates. These include but are not limited to:

- Good Manufacturing Practice (GMP) (pharmaceutical, including Active Pharmaceutical Ingredient (API), veterinary, and blood)
- Good Clinical Practice (GCP)
- Good Laboratory Practice (GLP)
- Good Distribution Practice (GDP)
- Good Quality Practice (GQP)
- Good Pharmacovigilance Practice

- Medical Device Regulations
- Prescription Drug Marketing Act (PDMA)

Harm

Damage to health, including the damage that can occur from loss of product quality or availability.

Hazard (ISO/IEC)

The potential source of harm.

Jailbreaking

The practice of breaking security protections to circumvent supplier restrictions on the use of a mobile device allowing root access to the operating system, and permitting the user to load apps not approved by the operating system supplier. It makes the device more vulnerable to both malware and to user-attributable malfunction, and typically voids the warranty and/or service contract.

Mobile Application (Mobile App) (FDA)

A software application that can be executed (run) on a mobile platform, or a web-based software application that is tailored to a mobile platform but is executed on a server.

Mobile Medical Application (Mobile Medical App) (FDA)

A mobile app that meets the definition of “device” in section 201(h) of the Federal Food, Drug, and Cosmetic Act (FD&C Act); and either:

- Is used as an accessory to a regulated medical device; or
- Transforms a mobile platform into a regulated medical device

Post-production Information

Safety relevant information gathered after product launch, including previously unrecognized hazards or changes in the risk profile.

Risk (ISO/IEC)

The combination of the probability of occurrence of harm and the severity of that harm.

Severity

A measure of the possible consequences of a hazard.

Stand-alone Software (MHRA)

Software which has a medical purpose which at the time of it being placed onto the market is not incorporated into a medical device.

Smartphone

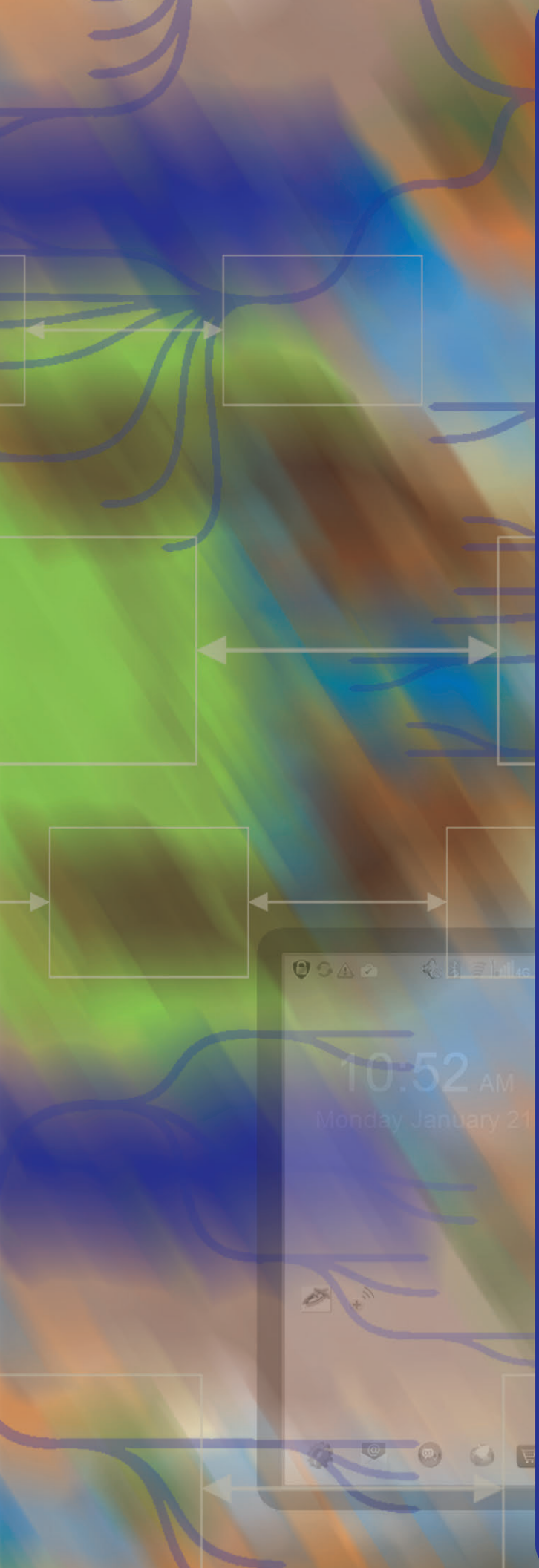
A mobile phone that is able to perform many of the functions of a computer, typically having a relatively large screen and an operating system capable of running general-purpose applications.

Tablet

A small portable computer that accepts input directly on to its screen rather than via a keyboard or mouse.

Unlocking

The practice of breaking security protections to circumvent supplier restrictions on the use of a mobile device, often for the purpose of using it on a competitor's network. While this is probably less risky than "jailbreaking" (see above) it could lead to communication or compatibility issues. Some unlocked devices are commercially available.



Connecting a World of
Pharmaceutical Knowledge

ISPE Headquarters
600 N. Westshore Blvd., Suite 900, Tampa, Florida 33609 USA
Tel: +1-813-960-2105, Fax: +1-813-264-2816

www.ISPE.org