Planning an information systems project

A TOOLKIT FOR PUBLIC HEALTH MANAGERS

OPTIMIZE

World Health Organization

PATH

THIS DOCUMENT WAS COMMISSIONED BY OPTIMIZE: IMMUNIZATION SYSTEMS AND TECHNOLOGIES FOR TOMORROW, a collaboration between the World Health Organization and PATH. The report was authored by Jan Grevendonk, Brian Taliesin, and Dan Brigden of PATH.

ACKNOWLEDGMENTS

The authors would like to thank all the people who provided valuable feedback during the creation of this guide.

In particular: Anup Akkihal, Thomas Cherian, Marta Gacic-Dobo, Andrew Garnett, Skye Gilbert, Ajay Goel, Heidi Lasher, David Lubinski, Henry Mwanyika, Sophie Newland, Liz Peloso, Kathleen Tiffay, Maeve Wagner, Allen Wilcox, Kate Wilson.

Design: Rebecca Richards-Diop

CONTACT INFORMATION:

Jan Grevendonk
Technical Officer, PATH
grevendonkj@who.int

Brian Taliesin
Technical Officer, PATH
btaliesin@path.org

Dan Brigden
Consultant
brigdend@who.int

World Health Organization
Avenue Appia 20
1211 Geneva, Switzerland
www.who.int

PATH

Mail
PO Box 900922
Seattle, WA 98109 USA

Street
2201 Westlake Avenue, Suite 200
Seattle, WA 98121 USA
www.path.org

ABOUT PROJECT OPTIMIZE

Project Optimize, a five-year partnership between the World Health Organization (WHO) and PATH, was established to identify ways in which supply chains can be optimized to meet the demands of an increasingly large and costly portfolio of vaccines.

Optimize worked directly with national governments and other institutions to identify problems in the supply chain and test innovative solutions. We also worked with vaccine manufacturers and policymakers to help ensure that new products and policies enable supply chain systems to function effectively. The goal was to help define an ideal vaccine supply chain that can be used to develop stronger, more adaptable, and more efficient logistics systems, extending the reach of lifesaving health technologies to people around the world.

For more information, please visit the Optimize websites.

PATH:
↗ www.path.org/projects/project-optimize

WHO:
↗ www.who.int/immunization_delivery/optimize

Cover photo: Joint Learning Network for Universal Health Coverage/Buddy Ganzone

World Health Organization, PATH. *Planning an Information Systems Project: A Toolkit for Public Health Managers.* Seattle: PATH; 2013.

This work was funded in whole or part by a grant from the Bill & Melinda Gates Foundation. The views expressed herein are solely those of the authors and do not necessarily reflect the views of the Foundation.

ABOUT THIS TOOLKIT

This toolkit can help public health managers to plan for the implementation of information and communications technology (ICT) in health information systems. It draws on lessons learned during project Optimize, a five-year partnership between the World Health Organization and PATH to help optimize the vaccine supply chain.

The toolkit focuses on the planning phase of an information systems project. It proposes an eight-step process that can help decision-makers:

- **Choose the solution that best fits their needs and context.**
- **Obtain the external help and expertise they need.**
- **Develop, scale, and then sustain their chosen solution.**

This document describes each step and includes planning tools to help you complete them. It is not an exhaustive technical guide to implementing information systems, however, and it does not address the strategic questions around a wider electronic health vision. Managers looking for more detailed guidance should refer to "**Finding more information**" on page 27 for additional suggested resources in these areas.

All health information systems use a variety of technologies that can include paper-based tools as well as ICT. In many cases, introducing new information technology presents the best opportunity to improve these systems. However, doing so also poses considerable challenges. This toolkit can help managers to meet these challenges by carefully planning their use of ICT.

There are eight steps in the process of planning an information systems project:

1.	DEFINE OUTCOMES	How will a better information system benefit you? How should you define the scope? How will you measure success?
2.	FORM YOUR TEAM	What skills and roles are required to bring your project to a satisfying outcome?
3.	DEFINE WHAT YOUR SYSTEM NEEDS TO DO	How can you define your requirements for the system?
4.	FIND THE RIGHT SOLUTION	Should you buy or build your system? Do you select an open-source or proprietary system? How do you evaluate different systems and select the best one?
5.	SELECT THE RIGHT VENDORS	How do you make sure you select the best providers of technical services?
6.	ESTIMATE IMPLEMENTATION AND OPERATING COSTS	How much will your project cost to pilot, scale, and maintain?
7.	CREATE AN IMPLEMENTATION PLAN	How long will it take to develop, pilot, and scale up?
8.	UNDERSTAND AND MANAGE PROJECT RISKS	What can go wrong and how can you plan for that?

TABLE OF CONTENTS

Click here on any page to return to the Table of contents

INTRODUCTION Why do so many health information systems fail?

Small-scale electronic health (eHealth) and mobile health (mHealth) projects and pilots are proliferating in Africa and Asia. Many of these are implemented by nonprofit or donor organizations in a small geographic area and with a narrow functional or programmatic scope (for example, a stock tracking and reporting system for malaria drugs in a particular district). Despite the interest of ministries of health and donor organizations to invest in information technology (IT), there are few success stories of widely implemented health information systems.

ACRONYMS

CRDM	Collaborative Requirements Development Methodology
eHealth	Electronic health
EU	European Union
ICT	Information and communications technology
IT	Information technology
LMIS	Logistics management information systems
mHealth	Mobile health
MOH	Ministry of Health
PATH	Program for Appropriate Technology in Health
RFP	Request for proposal
SaaS	Software as a service
TCO	Total cost of ownership

What is holding back the widespread adoption of information and communications technology (ICT) systems in the health systems of developing countries?

Why are so many seemingly great ideas never scaled up beyond the pilot phase?

Why do some sophisticated and proven systems bring little value or improvement to the health system?

There are many reasons, but typical barriers and pitfalls include the following.

Typical barriers and pitfalls

- Managers take shortcuts through established system development methodologies: they do not spend enough time at the beginning of the project on system analysis and design; therefore, the development time and effort escalates.
- The agendas of technical assistance providers and donors are not aligned with the interests of the system users.
- The architecture of the system is unsuitable for its envisioned scale and scope, or for the context in which it is deployed.
- There is a premature commitment to a fixed budget and schedule, with some project areas (usually those that are less well understood by planners and decision-makers) receiving insufficient resources.
- Pilot projects are not documented and evaluated well enough to demonstrate the efficiency gains, health outcomes, and value for money required to scale them up.

Unfortunately, many ministries of health lack the staff and general capacity to plan adequately for information systems projects. While this guide cannot fully address such shortcomings, it can help planners to address many of the challenges.

This document will focus on what could have been done instead: start with the larger vision and desired outcomes in mind; define the project scope well; then budget adequate time and resources for planning, requirements gathering, and system design.

Typically, there are three main phases of an ICT project: pilot, scale, and sustain.

Three ICT project phases

1 **Pilot**

The team develops or selects the right solution, based on program needs and priorities. The system is then tested in a small-scale pilot to measure outcomes, impacts, and costs and to identify potential improvements.

2 **Scale**

The system is deployed to its intended reach— for example, in all districts in the country, going from 50 to 2,000 users.

3 **Sustain**

To provide continued value, the system needs to be maintained and sustained. After five to ten years, it may need to be improved or even replaced, starting again with the first phase.

Quality assessment reviews should be planned between each phase. For example, the decision to move from development to national scale should depend on the successful conclusion of a pilot demonstration. Then, once the system has been scaled up, the outcomes should be evaluated against the original project objectives and, if necessary, design changes should be made. In the box to follow, we describe several real-life project failures and examine how better planning could have led to success.

EXAMPLES OF PROJECT FAILURES

Limited design

In the context of a technical knowledge transfer program sponsored by the European Union (EU), an EU country helped a middle-income country implement its immunization registry system. When the program ended, the receiving country's officials discovered they had no way to modify reports or functionality, or even access the database directly. They abandoned the system.

What went wrong?

It is likely that not enough time was spent planning and designing the project. It was simply assumed that what worked in one country would work in another. Many factors, not simply functionality, affect the feasibility and usefulness of a system in a specific country.

Not planning to scale

In one African country, a consortium of donors and technical partners implemented a short message service–based system to track the commodities used in a public health program. While it was scaled up nationally for some commodities, it will not be easy to extend its use to a large number of other commodities; sending a different text message for every transaction for a large number of commodities will be too cumbersome and costly.

What went wrong?

The system design was not aligned with the wider eHealth and mHealth vision. Instead, it focused on the demonstration of a technology. Failing to think through what would happen in the long run means that the system might not be adopted and maintained by the ministry of health of that country.

Risk management and implementation issues

Wealthy countries suffer project failures, too. The United Kingdom's "National Health Service National Programme for IT" was established in 2002 with the aim of implementing an electronic medical record system, among other things. The program was initially expected to cost £2.3 billion over three years, but in 2011 it was dismantled after having spent £6.4 billion without achieving its initial goals.

What went wrong?

There are probably many interlinked reasons, although an important one could be that it was too ambitious in scope and scale from the start, making it difficult to obtain the necessary engagement and support of stakeholders. A well-planned pilot phase with a well-managed scope would have limited the financial risk.

STEP 1: DEFINE OUTCOMES

Investment in information systems can help to strengthen national public health systems, but why should a ministry of health (MOH) with limited resources make such an investment when it has so many other priorities to consider? Making a clear case for information systems will help secure funding and prioritize the effort. Furthermore, defining the outcomes and scope of a project will focus stakeholders on common goals.

BY TAKING THIS STEP YOU SHOULD BE ABLE TO ANSWER...

- ✓ **What do I need my information system to do?**
- ✓ **How does an information system help us implement our strategy?**
- ✓ **How can I make a business case for investment in health information systems?**
- ✓ **Where is the system going to be implemented?**
- ✓ **How will I measure success?**

Identify problems and opportunities

A national health information system is the combination of people and IT that supports the operations, management, and evidence-based decision-making within a public health system. The ultimate goal of a health information system is to improve health outcomes by providing high-quality, relevant data in a timely manner.

Improvements to a health information system emerge from a need for change: either the existing system does not meet the evolving needs of the program, or there is an understanding that emerging technologies allow for radically better systems. Typically, a small group or "project champion" identifies this need and then tries to mobilize a bigger group of stakeholders. In this phase, the perceived gaps and opportunities are documented, with a strong focus on describing the desired benefits and the "why." It may include the development of a high-level business case that compares the benefits to estimated costs.

Think about which problems in your health system could be solved if you or other health workers had better (more accurate, complete, timely, or relevant) information. As possible, obtain quantitative data about health outcomes or efficiencies and measure whether the implementation of an information system could improve these indicators. A successful ICT project will produce many kinds of benefits for all of its users. Table 1.1 shows the kinds of benefits information systems can bring.

Table 1.1 The benefits a good information system can bring

BENEFIT	EXAMPLES	HOW TO EVALUATE SUCCESS
Better indicators for strategic planning	• Targeted advocacy efforts help to address higher-than-average vaccination dropout rates in specific population groups. • Credible estimates of vaccine wastage rates for each health center lead to tailored vaccination strategies to reduce wastage. • High failure rates in certain types of cold chain equipment lead to the discontinuation of that equipment.	• Did the system produce credible data for these indicators? • Were managers able to act on this information? • Did the information change decisions and how did that benefit the program? • Was there any measurable impact on outcome indicators, such as vaccination coverage?
Better day-to-day decision-making	• A district officer validates a vaccine request based on the available stock, target population, and average consumption in the health center that sent the request. • A nurse uses the immunization register of her clinic to find the children who are falling behind their vaccination schedule. • A warehouse manager analyzes average demand and makes sure that stock is kept between minimum and maximum levels.	• Did the system lead to operations that are more efficient? For example, was there a reduction in buffer stocks or wastage? • Did the system lead to better availability of stock? • Did it change the way people work and did that improve health outcomes (for example, higher coverage, lower dropouts)?
Better control and oversight	• In Senegal, some health programs have outsourced the distribution of their commodities to the national pharmacy. With access to stock and delivery information, they can regularly monitor the arrangement. • In Turkey, pharmacists scan barcodes when they dispatch drugs to make sure that the insurance system is not overcharged. • Through a last-mile stock management system, managers can monitor whether some health centers or districts are regularly overstocked or experiencing stockouts.	• Do the system data accurately reflect reality? • Did the system highlight poor performance?
Reduced administrative burden	• Health workers enter monthly reports directly into a computer or mobile device and transmit them electronically. • Aggregate coverage reports are generated automatically by the system.	• Comparing how people spent their time before implementation of the system change and how they spend it since the change, what are the differences?

CASE STUDY

ALBANIAN IMMUNIZATION INFORMATION SYSTEM

The Albanian Institute for Public Health worked to implement an immunization information system that digitizes child immunization records and the stock management system in an integrated manner. By recording every individual child, vaccination, and stock movement in a central database, the system is able to provide the following benefits:

1. Better estimates of immunization coverage.
2. Managers can list the registered children who are unvaccinated and use that list to find out why.
3. Reminders and automated work planning reduce dropout rates and improve the timeliness of vaccination.
4. When a bad lot needs to be recalled, managers can find out easily where it was distributed and to which children it was administered.
5. Ability to check whether new and more expensive vaccines are administered to the children they were meant for.
6. Nurses spend less time preparing for vaccination sessions and reporting to the district.

Photo: PATH / Ilir Kaso

The project champion should draft a business case that describes these benefits (either in quantitative terms or by assigning dollar values to each of them) and that compares the benefits to the estimated overall cost of implementing and maintaining the system. **Step 6** of this guide describes how to estimate the total cost of ownership (TCO) of a system. The business case needs to be refined after the pilot phase has confirmed the benefits. By being able to show value for money, you can increase your opportunities for internal and external fundraising.

Setting and respecting boundaries will be a key success factor for your project. You can achieve this by clearly establishing and agreeing on the scope—that is, **defining what is and is not included in your project**—in terms of its functional, programmatic, and geographic dimensions.

1. Functional

The functional scope of the system refers to what it does. Will it support patient records, disease surveillance, stock management, and logistics? Advocacy and outreach? Finance and accounting? Human resource management?

2. Programmatic

The programmatic scope refers to the public health programs that will be using the system. Is it going to be made specifically for one program or will it be integrated? To answer this question, it is worth thinking about the end users of the system. Will they end up needing several systems to do their job? In that case, their development needs should be coordinated.

3. Geographic

The geographic scope of the system refers to where it will be used, and by whom. Is it meant for nationwide deployment? At what levels of the health system? In hospitals? District offices? Community health centers?

CASE STUDY

LOGISTIMO IN SOUTH SUDAN

In South Sudan, the Expanded Programme on Immunization, United Nations Children's Fund, and project Optimize worked together to implement Logistimo, a stock management system that uses mobile phones. By allowing users to order vaccines and record transactions such as receipts, issues, and wastage in real time, Logistimo provided the following benefits:

1. Managers can view stock balances in remote locations, which helps to avoid stockouts and overstocking.
2. Managers can analyze consumption patterns to assess the appropriateness of order quantities.
3. Users can transmit orders over the phone and receive text message updates on when the shipments will arrive, allowing them to organize their work better.

Photo: PATH/Jan Grevendonk

You cannot deploy a system that does everything for everybody from the start. It makes sense to begin with a pilot, in which the scope is reduced to make the implementation more manageable and to allow you to find and correct weaknesses before you add functional, programmatic, or geographic scope. Often, this is achieved by testing the system first in one geographic area, and with a core set of the final functionality. After evaluation, it can then be scaled geographically, after which new layers of functionality are added over time, or it can be perfected and complemented with new functionality before it is implemented nationally. It is therefore important to have a sense of both the initial and final scope before you begin: the long-term vision will drive the system design and the architectural choices you make.

At the start of any major ICT project, the project management team should create a **project charter** or terms of reference document that summarizes what the project will accomplish and where and when they will implement it. That way, there is no confusion about the project's objectives, targets, and scope.

TOOLBOX

PROJECT CHARTER

An example project charter is provided at the end of this toolkit.

Annex 1. Project charter

STEP 2: FORM YOUR TEAM

There is more to IT projects than developing software; in fact, an information system is always a combination of people and technology. At the start of your project, you should form a multidisciplinary team that can manage every project function, not only the development, deployment, and operations functions, but also the management and governance functions.

To avoid misunderstandings and to make sure that your team possesses the required skills, consider the roles that will be required. Select individuals or organizations that will fulfill these roles. Completing a **roles and responsibilities matrix** can help.

BY TAKING THIS STEP YOU SHOULD BE ABLE TO ANSWER…

- ✓ What resources and skills do I need for each of the three phases (develop, scale, and sustain)?
- ✓ How can I assess what skills are already available in the MOH, other ministries, or partners?

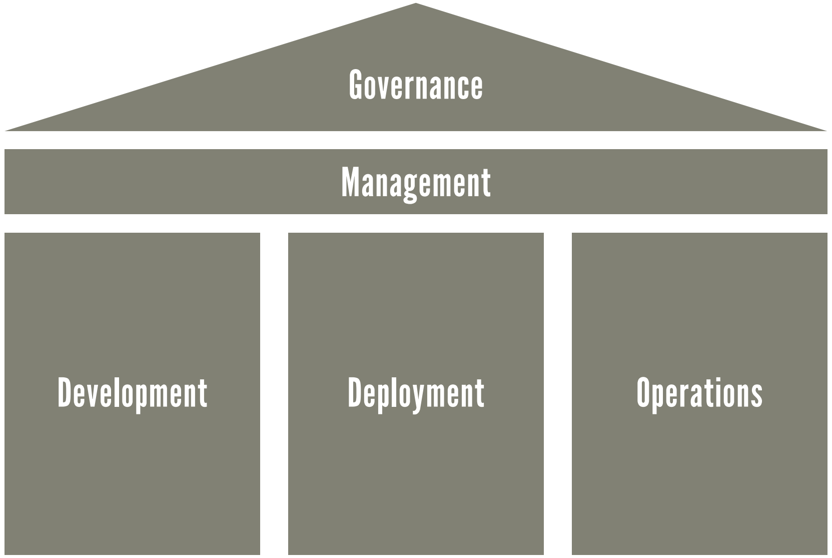


Figure 2.1 Project functions

Governance

There are likely to be numerous groups involved in your project, and each will have different objectives and interests. This can seriously complicate decision-making. Understanding the various perspectives and finding areas of consensus and collaboration are important to ensure the project's success. For projects of significant magnitude, a project champion from the senior level of the MOH who can align interests and resolve potential conflicts is required. This person should chair a committee of stakeholders that sets the overall direction for the system implementation, makes key decisions, and identifies key team members. Some typical views of stakeholders in an information systems project are given in Table 2.1.

Table 2.1 Views of stakeholders in an information systems project

GROUP	OBJECTIVES + INTERESTS
Ministries of health and finance	Improved outcomes from investment. Project implemented on time, within budget, and meeting user requirements.
Ministry or department of IT or planning	Alignment of project with the national eHealth strategy and compliance with policies. Ensuring the project leverages existing investments in IT servers, communication networks, etc.
Funding sponsor	Lasting impact and demonstrated value for money of the project.
Users	Delivery of a system that meets their requirements with acceptable usability, performance, and flexibility.
Project team	Meeting short-term criteria set by project sponsors and the funding organization.
Subcontractors	Receiving short-term payments for services delivered.
Vendors	Establishment of a long-term revenue stream.
Researchers	Validation of hypotheses. Publication submissions.

STEP 2: FORM YOUR TEAM

Management

The project needs to be planned, managed, monitored, and evaluated by a project manager and support staff. This management group should also include administrative functions such as procurement. Often, more than one project manager oversees a project: typically one representing the MOH, one representing the donor organization or technical agency that supports the project, and one designated by the IT company that has been selected. Having multiple project managers may be acceptable as long as roles and responsibilities have been clearly defined and the MOH project manager is in control of the overall direction of the project.

Development

The development or customization of a software system is a specialized and time-limited role. Therefore, the development team normally needs external resources, such as IT contractors or technical assistants. The development function is most important in the pilot phase. However, during the scale-up phase, the development team will still need to fix any software problems and accommodate requirements not identified during the pilot phase. Even during the maintenance phase, development work may still be necessary, for example, because users require new reports. The skills in this team vary from very technical (such as database administrator) to more analytical (business analyst, for example).

Deployment

To implement the system at scale, a new set of skills is required, mostly around training and the mass deployment or upgrade of computer equipment. This function may also rely on short-term technical assistance.

Operations

Once implemented, the system needs to be maintained, and users will need some form of overall support. This is where the operations team comes in. Key skills include database management and help desk support.

TOOLBOX

ROLES AND RESPONSIBILITIES MATRIX

An example roles and responsibilities matrix is provided at the end of this toolkit.

Annex 2. Project roles and responsibilities matrix

STEP 3: DEFINE WHAT YOUR SYSTEM NEEDS TO DO

In a typical project, public health staff work together with technical people such as vendors, developers, consultants, and IT staff in the MOH. Often, these two groups have completely different backgrounds and even seem to speak different languages. As a result, a lack of understanding between the future users of a system and the people who design it is one of the main causes of frustration, delays, cost overruns, and even failures in IT projects.

BY TAKING THIS STEP YOU SHOULD BE ABLE TO ANSWER...

- ✓ **How can I document my system requirements and communicate better with technical people?**
- ✓ **Which processes will be needed to support the objectives and outcomes?**
- ✓ **What are the technical requirements I should be worrying about? How will the information system perform when we have a problem with electricity and/or Internet connectivity?**
- ✓ **How does the design of my health information system enable our country to maintain it? Is it possible to design an information system so that training is minimized?**
- ✓ **How will newly computerized processes link to retained manual and paper-based processes?**

A methodology to document user requirements can help to mitigate the risk of misunderstanding and ensure that program managers remain in control of system functionality. One such methodology is the Collaborative Requirements Development Methodology (CRDM) developed by the Public Health Informatics Institute and PATH. The CRDM is a methodology for low-resource settings that enables public health managers to define and document their requirements in a standardized and collaborative way.

A methodology by itself is not sufficient—true understanding depends on people. The bridging function between different groups is often filled by a business analyst—a public health person who understands ICT or an ICT person who understands public health. Focusing too much on current processes may also lead to lost opportunities to transform and improve them. Resources permitting, experts in human behavior change can help to ensure that the system is built in a way that makes people want to use the data it produces.

Defining business processes can encourage you to think about the way people work (or could work) before you dive in to the details of what your system will look like. This process enhances the likelihood that the system will adapt itself to people and their work objectives.

Table 3.1 Example process matrix

#	PROCESS	OBJECTIVE	INPUT	OUTPUT	TASK SET	OUTCOMES
A	**Patient management**	Maintain a database of all newborns, with their vaccination history.	• Patient information from hospital information systems • Newborn information from civil registry systems	Patient records available for other processes in the application	• Register patient • Search for existing record • Maintain patient database	More complete registration of newborns
B	**Vaccination management**	Ensure that all infants are vaccinated with all vaccine doses in the national schedule.	• Patient records from process A, "Patient management" • National immunization schedule	Vaccination records	• Define national schedule • Plan vaccinations • Send reminders • Register vaccinations • Monitor vaccination coverage	More timely vaccination, higher vaccination coverage
C	**Stock management**	Ensure that vaccines and other stock are always available when needed, while minimizing wastage and excess stock.	• Historic vaccine usage data from process B, "Vaccination management" • Stock available?	Data about available lots for process B, "Vaccination management"	• Order stock • Receive • Store • Count stock • Monitor balances, expiry dates, wastage, and usage	Higher availability and lower wastage of vaccine and other stock
D	**Cold chain management**	Ensure that high-quality cold chain equipment is available where needed.			• Register equipment • Update working status • Monitor working status	Higher uptime of cold chain equipment
E	**Reference data management**	Provide common reference data for other modules and processes.		Reference data	• Define health facilities • Define system users • Define stock items • Define geographies	Reference data available for other systems/ processes/ modules

Describe processes

A business process is a set of tasks that together accomplish a goal or produce something of value for the benefit of the organization.

An information system aims to make business processes more effective or efficient, and a detailed process analysis is the starting point to define what the information system is supposed to do. Business processes can be supported by manual as well as automated activities, and by paper-based as well as digital information.

The process matrix in Table 3.1 above includes illustrative processes for an immunization information system. Completing a process matrix such as this can help you with your work, and should be adapted for your situation. Thinking through the process will help you identify the details that are critical for your context.

Create task flows

A task flow diagram describes the activities of a business process and the people who perform those activities. The task flow provides a "story" for the process and serves as a focal point for achieving clarity and agreement among group members and stakeholders. Task flow diagrams look like standard flow charts.

Figure 3.1 Collaborative requirements session

Photo: PATH/Henry Mwanyika

Figure 3.2 Example task flow diagram for drug dispensing

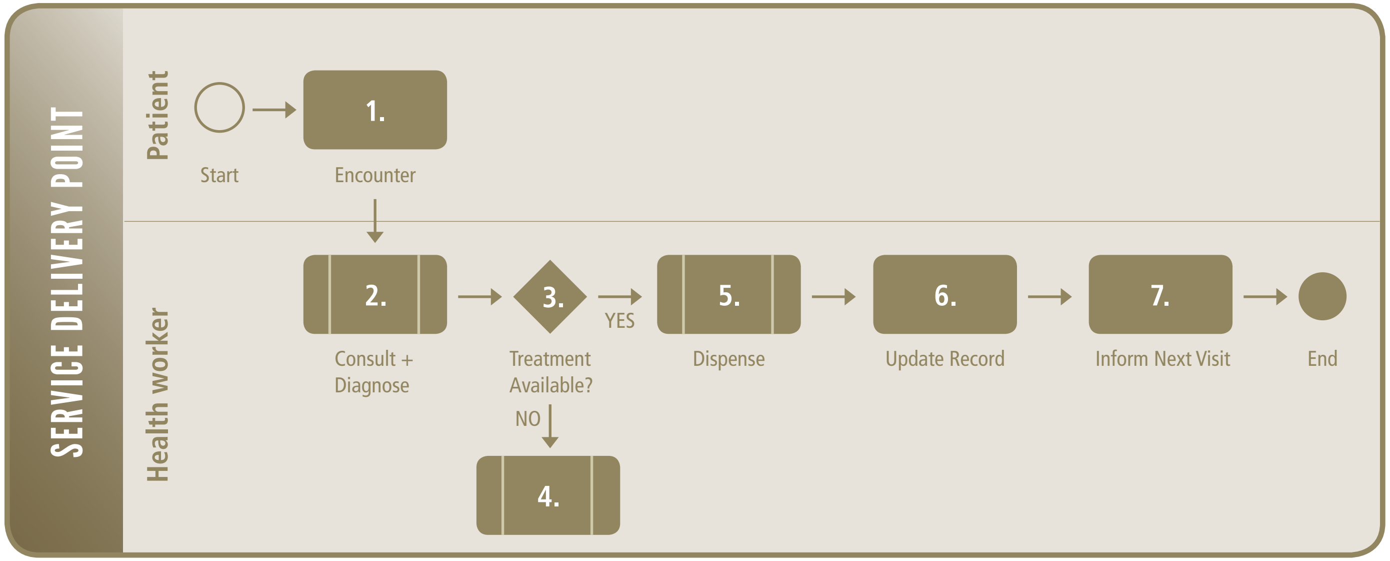


Figure 3.1 shows technical officers discussing, describing, and improving a set of task flows. Organizing a workshop like this and drawing on diverse resources from all levels (health center and district staff, as well as national and global experts) can help to ensure that every issue is considered from different points of view before the developers start work. Another benefit is that participation in such workshops creates a sense of investment and ownership in the project.

Task flow diagrams have a start and an end, and are read from left to right or top to bottom. Rectangles are used to describe the steps. Diamonds indicate decisions, and arrows indicate the sequence. They also specify who is responsible for the step and where it is taking place. An example task flow diagram is shown in Figure 3.2.

Define requirements

Requirements specify what is needed to solve a problem or to achieve the objective. Requirements may also be rules required within the system to satisfy a contract, standard, or specification. For the purpose of this toolkit, we distinguish functional and nonfunctional requirements.

Functional requirements

Functional requirements are usually expressed as statements that begin with, "The system must or should…" They express the functional abilities of a system, for example, the ability to generate a certain report or the ability to keep track of individual lots of a vaccine. An example of a user requirements matrix is shown in Table 3.2. This list can be used:

- To evaluate an existing system (does it comply with the requirements?)
- To communicate requirements to a developer who can find the best way to implement them.
- As a checklist to test a system that was developed.

Nonfunctional requirements

Nonfunctional requirements often describe the technical and environmental constraints that vendors and developers need to understand, such as the availability of electricity or Internet connectivity in certain areas.

TOOLBOX

NONFUNCTIONAL REQUIREMENTS

A checklist for nonfunctional requirements is provided at the end of this toolkit. Annex 3. Nonfunctional requirements

CRDM TOOLS

For more information on how to define processes, task flows, and functional requirements, please refer to the following document: Common Requirements for Logistics Management Information Systems

Table 3.2 Example user requirements matrix

PROCESS: VACCINATION MANAGEMENT **CONTACT:** STEVEN SMITH

PROCESS	MUST?	THE SYSTEM MUST OR SHOULD
1	✓	Allow the user to define a national vaccination schedule. This schedule defines the ideal age at which children should receive a certain vaccine dose within three constraints: • The minimum age before which a child is not eligible for a certain dose. • The maximum age beyond which a child is not eligible for a certain dose. • A minimum time that needs to pass between doses of the same vaccine.
2	✓	Reproduce (display and print) a list of children requiring immunizations from a certain health center at a particular time, with all the doses that are due and overdue.
3		Send SMS (short message service) text message reminders to caregivers.
4	✓	Allow users to update a child's immunization record with the date of vaccination, the health center where the vaccination took place, and the vaccines and doses that were administered.
5		Register the lot number that was used for each vaccination in a child's immunization record.
6	✓	Reproduce (display and print) a child's vaccination history, together with all due and overdue appointments.
7	✓	Reproduce (display and print) a coverage report that shows vaccination coverage as the percentage of children living in a certain area that were born within a certain time frame and that were vaccinated with a certain vaccine dose.
8	✓	Reproduce (display and print) a report that shows all vaccinations administered— by dose and by health facility or group of health facilities (district, region, or national level).

STEP 4: FIND THE RIGHT SOLUTION

Now that you know what you want to achieve (Step 1), have formed a team (Step 2), and defined what your system will do (Step 3), you need to find the best solution for your problem and context. In this step, you will find an appropriate system solution that aligns with your context and national information architecture.

BY TAKING THIS STEP YOU SHOULD BE ABLE TO ANSWER...

- ✓ **What is an enterprise architecture and why does it matter?**
- ✓ **What are others doing?**
- ✓ **Should we build, buy, or adopt a system?**
- ✓ **What is open-source software, and software as a service (SaaS)?**
- ✓ **Is open-source software always a better alternative than purchasing from a vendor?**
- ✓ **How do I compare various options objectively?**

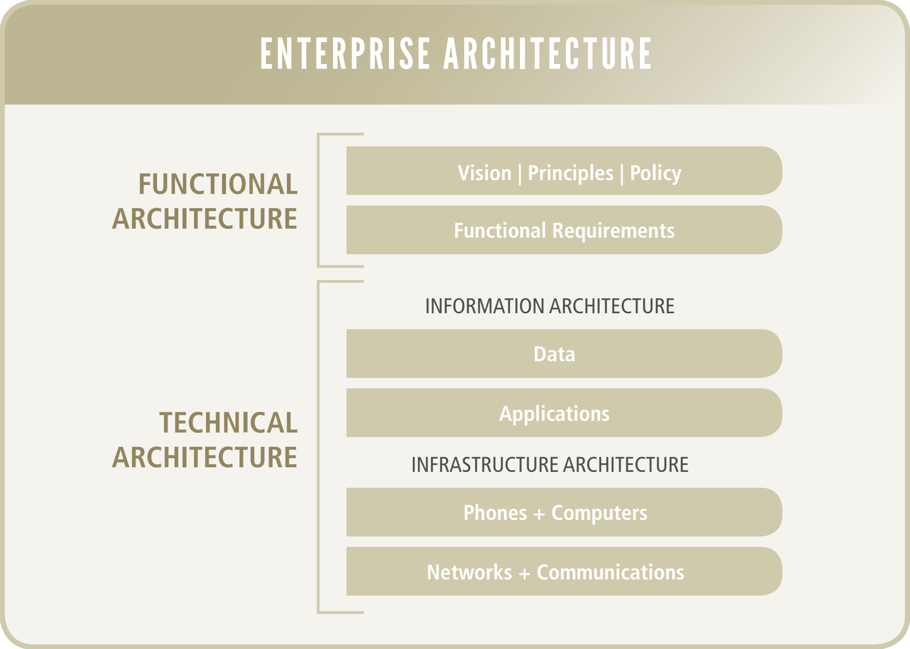
Consider existing systems and standards in your country

Before looking at potential solutions, you should consider the existing systems, standards, and policies in your country. Are there any other systems the new system will need to link with? What kind of equipment and technology is already in use and where? Does any authority mandate compliance with certain standards for data and communication? Are there any policies regarding data security and privacy? If so, this will focus your options. Ideally, any existing policies and standards will be described in a national eHealth strategy document. In this case, you must be sure to align your choice of solution with these requirements.

The standards and policies to be followed can be conceptually represented in an enterprise architecture like the model in Figure 4.1. In broad terms, the architecture defines three areas.

- **INFRASTRUCTURE ARCHITECTURE:** the hardware to be used and the network and data exchange protocols to be followed.
- **INFORMATION ARCHITECTURE:** the data and application standards to be followed, which include the mandatory use of existing reference tables (for example, a list of medical products) or a common service application (for example, a vital registration system that needs to be used by other applications).
- **FUNCTIONAL ARCHITECTURE:** the national vision, principles, and information policies, including those on privacy and data security. It can also include the repository of functional requirements for health applications.

Figure 4.1 Enterprise architecture model



Annex 5 lists important governance and design principles that robust information systems should adhere to.

In the absence of a national eHealth strategy, decision-makers are often unable to align their system with the existing systems, standards, and policies in their country, and choose instead a health application to meet their immediate and segmented needs. This creates a situation that is difficult for the MOH to manage and sustain.

Find out what others are doing

Too often, the wheel is reinvented when it comes to developing software. Even if you are convinced that you need a system that is custom developed for your needs, it still makes sense to look at what other people are doing, in a different country or in a different program in your country. This is not always easy because there is no single repository that contains many experiences around information systems projects in public health. Furthermore, information is most often advocacy oriented, and biased toward success stories. Not much documentation exists regarding challenges, lessons learned, or technical details. However, we have provided details of some resources that are available in "**Finding more information**"on page 27.

Explore different software models

Before deciding on a specific software solution, consider the advantages and disadvantages of the different strategies that exist to acquire a software system: you can build a system from scratch or choose something that already exists. This can be a commercial application, or a freely available system. Table 4.1 provides an overview of the available options.

Table 4.1 Benefits and risks of different software models

MODEL		BENEFITS	RISKS
Custom-developed software Build a software system from scratch.	**Examples:** Project Optimize demonstration projects in Albania, Guatemala, Senegal, and Vietnam.	• You have control over technology, functionality, and design. • The development experience creates ownership and improves sustainability. • It is possible to engage the local IT industry.	• Custom development tends to be difficult to manage within time and budget. • Control over design does not guarantee satisfaction with the end product, as that depends on the capabilities of the technical team. • Long-term support depends on the continued availability of individuals.
Commercial off-the-shelf software Buy a commercially available product.	**Examples:** Sage Enterprise Resource Planning, which is in use in many countries in Francophone Africa for essential medicines.	• The lead time from selection to implementation is normally shorter. • You can evaluate it before buying. • The product is maintained and upgraded (at a cost). • It has normally been tested and refined in other implementations.	• Often expensive and sold with unclear and complex fee structures, for example, a fee-per-server processor. • Commercial off-the-shelf software is not often designed for implementation in low-resource settings.
Free packaged software Software developed by a donor organization or technical agency. Alternatively, a system developed by a neighboring country.	**Examples**: USAID/John Snow, Inc.: • PipeLine • Supply Chain Manager World Health Organization: • Vaccination Supplies Stock Management tool • District Vaccine Data Management tool	• Shorter lead time. • Possibility to evaluate. • No upfront cost (but maintaining or customizing it may require investment).	• There is often no contract, so service and warranty for bug-fixing depends on goodwill of one or two individuals and there is no institutional support. • Many implementation and running costs are hidden.
Open-source software The source code as well as the software product is freely available. Often, a community has been formed to support the open-source software.	**Examples:** OpenLMIS.org OpenMRS.org DHIS2.org OpenXData.org	• You have the right to make changes to the software. • You can engage the local IT industry. • Benefit from communities and share development costs with other organizations.	• Can end up with a poorly supported product. • A loosely knit community might not be able to provide the business relationship you need. • Some of the implementation and running costs are hidden.
Software as a service (SaaS) Database and application hosted on remote servers, and software is sold (or offered freely) as a service that can be contracted per user and per month or year.	**Examples:** Logistimo Magpi	• Highly feasible to implement and maintain. • Clarity about the cost to implement and run a SaaS application. • Investment in improved software can easily be shared among customers.	• Data hosted on remote servers: not always in agreement with national policy. • Ministries of health are not often well positioned to pay a regular service fee.

Select a solution

The choice of a software system depends on two factors:

1. **The match between the proposed solution and the requirements you have defined**
2. **The total cost of ownership (TCO) of the system**

The acquisition or development cost is a part of the TCO, but even more significant are the scale-up and maintenance costs. In this context, the user-friendliness and ease of implementation of the system are crucial. Sophisticated, full-featured systems may look attractive, but implementing them may be costly. In other words, do not buy a Ferrari if you are going to drive it on dirt roads. See **Step 6** for a more detailed discussion of the impact of design on cost.

TOOLBOX

SELECTION MATRIX
An example selection matrix is provided at the end of this toolkit. This can help you to select the best option from your list of possible solutions. Annex 4. Selection matrix

GOVERNANCE AND DESIGN PRINCIPLES
A list of important governance and design principles that robust information systems should adhere to is provided at the end of this toolkit.
Annex 5. Governance and design principles

FIND OUT WHAT OTHERS ARE DOING
The USAID | DELIVER PROJECT conducted a supply chain software survey in 2012 that includes information on 54 software applications used at different levels of the supply chain in 30 countries.

STEP 5: SELECT THE RIGHT VENDORS

It is easy to be overwhelmed by vendors with more experience selling their service or product than you have buying it. This step helps you to ask the right questions when selecting a vendor.

BY TAKING THIS STEP YOU SHOULD BE ABLE TO ANSWER…

- ✓ How can I select the right vendors and consultants?
- ✓ How do I ensure that a contract for an information system clearly specifies how the system should perform?
- ✓ How will I ensure that short-term technical consultants will provide my team with the knowledge and skills I need to sustain my information system?

Issue a request for proposal

In many cases, the MOH and technical assistants do not have all the required skills to implement an information system successfully. You will therefore need to engage with one or more vendors of specialized ICT services or equipment. An important first step in doing this is to issue a request for proposal (RFP). An RFP is a document that is distributed to potential vendors requesting them to provide details of the product or services they intend to provide. To assist the vendor in submitting an RFP, you will need to provide detailed information on what you expect the vendor to deliver. The **functional and nonfunctional requirements** from Step 3 are good elements to include in an RFP, as they clearly communicate what you expect from a system. The RFP should also explain how you will evaluate proposals.

If you will select multiple vendors, it is important to be clear on how they will work together. A good practice is to ask one vendor to take the overall lead and responsibility, and subcontract with other vendors.

At a minimum, the RFP should state that proposals must include:

- A description of the proposed solution.
- A description of how the solution will be implemented in your context.
- An implementation work plan with timeline, methodology, roles, and responsibilities.
- The technical and organizational capabilities of the vendor, highlighting past projects that are most relevant for this work.
- The cost and level of effort, including the effort required from MOH staff.

Evaluate the merit of each proposal

A core team should screen proposals before a procurement committee evaluates them in depth. Having broad expertise on a procurement committee can help to ensure that key aspects of a proposal are not missed. This expertise should include subject matter experts in the system domain (for example, immunization systems, logistics, and laboratory systems, among others), system architecture, business analysis, project management, software development, procurement, and senior management.

The procurement committee is responsible for evaluating proposals based on the evaluation criteria outlined in the RFP. The committee shortlists two or three proposals, and after demonstrations, in-person interviews, and vendor presentations, a winner is selected. The goal of in-person interviews is to clarify any doubts and to assess whether a good rapport exists between the individuals that will be working on the project.

Evaluate the cost of each proposal and select a vendor

Next, evaluate the cost of the shortlisted proposals. This is a difficult matter because you should consider not just the contracting costs for this vendor, but also any implication on the overall lifecycle costs of the system. It may be justified to select a proposal or system with a higher upfront cost if the maintenance costs are expected to be lower, if the solution is technically superior, or if the delivery date is more favorable.

Make a contract or memorandum of understanding

Finally, award a contract to the winning vendor. If you will not have a commercial relationship with the selected technical partner (for example, because it is a non-governmental organization or university that receives funding from a different source), you still need to formalize your collaboration in a memorandum of understanding. The contract or memorandum of understanding may refer to the RFP or contain its main elements. It clearly describes the expected deliverables and stipulates milestones (the dates when key outcomes and deadlines will be met). Failing to meet these milestones may lead to deferred or reduced payment. The contract should also deal with licensing (who will own the system, including the software code?) and service after the project (until when will the company be required to fix bugs with no extra payment?).

TOOLBOX

PROPOSAL SELECTION MATRIX

This matrix will help you evaluate multiple received proposals in a systematic and standardized way. An example matrix is provided at the end of this toolkit. **Annex 6. Proposal scoring matrix**

VENDOR QUESTIONS

It is not always easy to ask the right questions from technical vendors. A list of suggested questions is provided at the end of this toolkit. **Annex 7. Vendor questions**

STEP 6: ESTIMATE IMPLEMENTATION & OPERATING COSTS

Most people underestimate the costs related to information systems projects, and often focus only on the costs related to initial development and piloting. Understanding the TCO is important to understand the funding needs and the long-term impact on the program budget.

BY TAKING THIS STEP YOU SHOULD BE ABLE TO ANSWER…

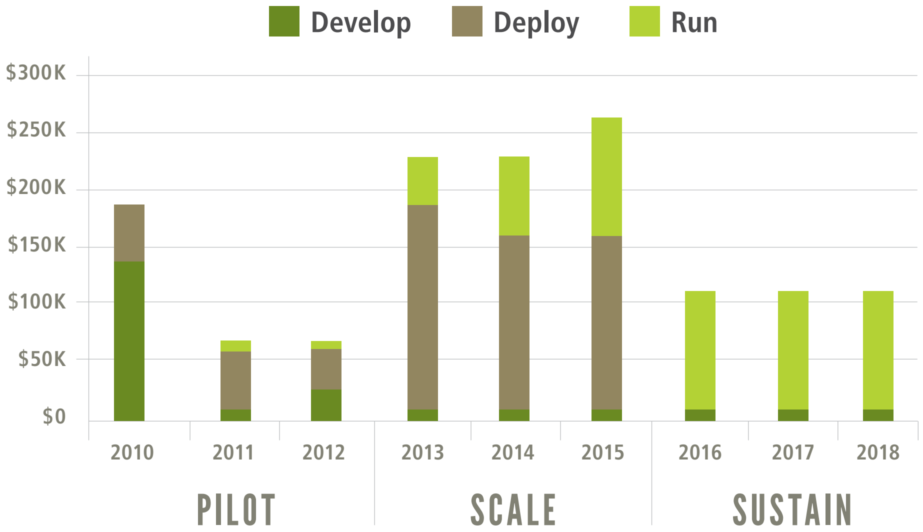
- ✓ **How much will this information system cost to develop, scale, and sustain?**
- ✓ **What are the individual major cost categories and the variables that drive these costs?**
- ✓ **How do I budget for information system support and maintenance?**

Calculate the total cost of ownership

The TCO includes not just the initial investment, but also the costs to scale and sustain the system over three to five years after implementation. Looking at the TCO helps to drive greater investments in certain aspects of the system. For example, an initial investment in improved usability may reduce long-term training and support costs.

Figure 6.1 below shows a hypothetical but representative cost profile that clearly highlights that the largest costs will not be incurred in the development phase, but in the consecutive scale and sustain phases of the system's lifecycle.

Figure 6.1 Cost profile for a typical information system



Understand the cost drivers

The World Health Organization and International Telecommunication Union estimate that eHealth projects typically allocate 60 to 70 percent of the budget to human resources (with an emphasis on training). Approximately 10 to 15 percent is spent on additional equipment, another 10 to 15 percent is spent on supporting ongoing operations, and the remaining amount is spent on other aspects of the project. Based on the scale of the system and the existing ICT infrastructure, the percentages may vary.

During different phases of the project, from development to deployment and operational support, the spending on various cost categories changes:

- **PILOT:**
  The functional, technical, and organizational complexity of the project drives costs. Costs do not vary significantly for a large or a small country.
- **SCALE:**
  The number of future users and the cost per user to deploy it are the most important variables. The cost per user depends on the way in which users will access the system (for example, desktop computer, mobile phone, paper) and their training needs.
- **SUSTAIN:**
  Apart from the number of users, the selected technology is critical here. For example, any solution that requires local software installation and maintenance will be more expensive than a centralized system, such as a web-based or cloud system.

The cost to maintain the system should provide an ongoing benefit that outweighs this cost. When seeking to manage the cost of operations, the organization should review ways to reduce the cost of:

- Data and communication plans.
- Software and hardware maintenance fees.
- Hardware replacement.
- Ongoing training needed due to staff turnover and refresher training.

TOOLBOX

DETAILED COST DRIVERS

A more detailed discussion of cost drivers is provided at the end of this toolkit.

Annex 8. What drives costs in all phases of the project lifecycle?

BUDGET MATRIX

A budget matrix can be used to summarize the main spending buckets and allocation across categories. An example budget matrix is provided at the end of this toolkit.

Annex 9. Total cost of ownership budget matrix

STEP 7: CREATE AN IMPLEMENTATION PLAN

Just as costs are often underestimated, so is the time it takes to implement a successful information system. This sometimes leads to hurried analysis phases, unclear requirement and scope definitions, and skipping necessary quality controls. In the end, this results in even longer timelines. Having a plan and following a project methodology is the key to ensuring success. This step highlights some attention points.

BY TAKING THIS STEP YOU SHOULD BE ABLE TO ANSWER…

- ✓ What are the key tasks necessary for successful deployment?
- ✓ How do I break the project into discrete milestones so that we demonstrate value and progress to all stakeholders?
- ✓ What are the main quality control points?
- ✓ How do I monitor whether the project is meeting its objectives?
- ✓ What role should I play during the implementation?

Define a work plan

The key tasks in the implementation plan vary by the kind of project and by project phase. **Annex 10** provides a list of activities that may be a good starting point to assemble your own work plan. Work plans are often finalized in collaboration with selected contractors or technical partners.

Track milestones

The project manager will divide the project phases into discrete milestones to show and report progress. The project management team should monitor the timeline, budget, and quality, as well as manage risks.

The project manager handles much of the responsibility for tactical details. However, senior leaders need to remain involved in the following ways:

- Participate in regularly scheduled meetings to review project status.
- Coordinate partners and facilitate collaboration at all levels of the health system.
- Provide oversight for issues that cannot be resolved by members of the project team, maintaining the focus on the project goals and scope.
- Hold team members accountable for their roles and responsibilities, ensuring that the various departments and components of the health system support project activities in good faith and adhere to work plans and schedules as determined in their individual work plans.
- Assess team performance and ensure that the organization takes ownership of operations.

Typically, developers will work in short iterations to develop a software system and organize informal tests as they do so. However, it is a good idea to organize at least three formal signoff points:

1 **User acceptance test**

A functional user acceptance test in which key users test whether the system lives up to their requirements. At a first stage, this takes place in a controlled central setting, with test data. (Sometimes this is referred to as a "conference room pilot.")

2 **Pilot test**

A pilot test in a real-life setting in which a limited number of real users work with the system, entering real data and following their normal process as much as possible.

3 **Volume (stress) test**

A volume or stress test, in which a large number of users enter a large number of transactions to ensure the system is fit for a larger scale.

TOOLBOX

WORK PLAN

An example work plan is provided at the end of this toolkit.

Annex 10. Work plan

STEP 8: UNDERSTAND AND MANAGE PROJECT RISKS

Following the seven steps described previously should lower project risk by aligning requirements to organizational objectives, understanding costs, planning appropriately, and choosing the right vendors and partners. However, despite the planning that takes place, a significant number of projects have a different outcome than expected.

BY TAKING THIS STEP YOU SHOULD BE ABLE TO ANSWER…

- ✓ What are some critical factors for project failure?
- ✓ How can project risk be managed?

In addition, low adoption by the intended users often prevents the system from achieving the intended results. Perhaps most concerning are projects that continue to invest resources and political capital despite evidence showing they have a limited chance of success. What are the common risks for failure in a project and how can they be managed?

COMMON RISK FACTORS

A risk factor is a situation that may give rise to one or more of the following project risks:

Lack of governance

Projects that do not have active participation or adequate representation by senior leadership often struggle. In fact, many cite lack of leadership buy-in as the most important factor for project success.

Poor management

The management team lacks the technical capacity or the organizational authority to provide the project the stability it needs.

Development risk

Relates to changing user requirements and a misunderstanding of the technology that is being used.

Deployment risk

Stems from a failure to manage the changes that will affect the organization because of the new information system.

Operational risk

Arises when the organization is not ready to support newly introduced technologies over the longer term.

Managing risk

To manage your project effectively, you need to be able to understand and manage risk. Understanding risk is the first and most important step in this process. Without doing so, you will not be able to minimize or even eliminate its impact on your project. Once you have understood the nature of a risk, there are four ways to manage it.

FOUR WAYS TO MANAGE RISK

1 **Avoid risk**

The team develops or selects the right solution, based on program needs and priorities. The system is then tested in a small-scale pilot to measure outcomes, impacts, and costs, and to identify potential improvements.

2 **Transfer risk**

The system is deployed to its intended reach, for example, in all districts in the country, going from 50 to 2,000 users.

3 **Mitigate risk**

To provide continued value, the system needs to be maintained and sustained. After five to ten years, it may need to be improved or even replaced, starting again with the first phase.

4 **Accept risk**

Sometimes nothing can be done to prevent the risk; instead, the risk consequences should be part of the plan.

A checklist including the most common risk factors in the implementation of health information systems is shown in Table 8.1 below. An extra column is provided for project managers to enter possible strategies to mitigate the risk.

Table 8.1 Risk management checklist

RISK FACTOR	IS IT A RISK?		MITIGATION STRATEGY
Governance			
Lack of a project champion/support from upper management	YES ○	NO ○	
Misalignment of partners’ objectives and stakes	YES ○	NO ○	
Political games/conflicts	YES ○	NO ○	
Unreliable external partners	YES ○	NO ○	
Organizational instability	YES ○	NO ○	
Management			
Changes to membership on the project team	YES ○	NO ○	
Lack of project leadership	YES ○	NO ○	
Lack of required management knowledge or skills	YES ○	NO ○	
Lack of clear role definitions	YES ○	NO ○	
Large and complex project	YES ○	NO ○	
Increased scope of project	YES ○	NO ○	
Changes to requirements	YES ○	NO ○	
Insufficient resources	YES ○	NO ○	
Development			
Introduction of a new technology	YES ○	NO ○	
Unreliable technical infrastructure or network	YES ○	NO ○	
Complex software solution	YES ○	NO ○	

RISK FACTOR (CONTINUED)	IS IT A RISK?		MITIGATION STRATEGY (CONTINUED)
Development (cont)			
Complex/incompatible hardware	YES ○	NO ○	
Poor software performance	YES ○	NO ○	
Poor perceived system ease of use	YES ○	NO ○	
Poor perceived system usefulness	YES ○	NO ○	
Misalignment with local practices and processes	YES ○	NO ○	
Deployment			
Unrealistic user expectations	YES ○	NO ○	
Overall resistance to change	YES ○	NO ○	
Lack of cooperation/commitment from users	YES ○	NO ○	
Lack of computer skills and knowledge among users	YES ○	NO ○	
Prior negative experiences with information and communications technology projects	YES ○	NO ○	
Operations			
Lack of local personnel knowledgeable in information and communications technology	YES ○	NO ○	
Lack of predictable long-term funding	YES ○	NO ○	

FINDING MORE INFORMATION

Public health managers planning the implementation of health information systems may find the following resources helpful.

GUIDES TO PLANNING HEALTH INFORMATION SYSTEMS

National eHealth Strategy Toolkit

Created by the World Health Organization and International Telecommunication Union, this toolkit is an expert guide for the development and implementation of a national electronic health (eHealth) vision, action plan, and monitoring framework.

↗ **www.who.int/ehealth**

Computerizing logistics management information systems

Created by the USAID: DELIVER PROJECT, this excellent resource includes detail around development and implementation, including some of the concepts contained in this toolkit.

↗ **http://deliver.jsi.com/dlvr_content/resources/allpubs/guidelines/GuidImplCLMIS.pdf**

ONLINE RESOURCES

TechNet-21.org

TechNet-21.org is a technical network for strengthening immunization services. This is a community-based forum and sharing network, moderated by the World Health Organization, and mostly used by logisticians at global, regional, and country levels. It contains an increasing number of posts about information systems in immunization programs.

↗ **www.technet-21.org**

OpenLMIS.org

OpenLMIS.org is a collaboration of domain experts in logistics and supply chains, eHealth information systems, software development for low-resource settings, and process improvement. Like other open initiatives, the intention is to ensure that OpenLMIS becomes a place for sharing information about logistics management information systems (LMIS) planning, requirements, and system design; promoting interoperability between systems; and developing open-source solutions for effective, scalable, and sustainable results.

↗ **www.openlmis.org**

HINGX.org

HINGX.org (Health Ingenuity Exchange) is a community-driven platform that brings together reusable tools, guidelines, and personal experience in health information systems. It facilitates the productive exchange of such material in a way that propels the development of health information technology. It is aimed primarily at eHealth and information and communications technology professionals.

↗ **www.hingx.org**

ANNEX 1. PROJECT CHARTER

An example project charter is provided below. This should be completed by the project management team to summarize what the project will accomplish and where and when they will implement it.

Vision/objectives

A concise description of what outcomes are expected from the system. Describe how the organization will benefit at the end of the project.

Background

Current situation that requires a system change; inventory of existing tools and systems; context diagram that visually represents the project participants, problems, and opportunities.

Functional scope

A brief description of the main functional blocks or modules that will be included.

Programs to be included

Which of the Ministry of Health departments and programs will eventually use this system? Will it include only a subset at first, and then be expanded?

Geographic scope

Where will the system be implemented over time? Where will it be piloted? Who will be using it? District people or also at the health center level?

Participants

List of individuals whose input has been gathered as part of the scope definition.

Timing

By when do you expect the system to be operational at the pilot level? And at scale?

ANNEX 2. PROJECT ROLES AND RESPONSIBILITIES MATRIX

An example roles and responsibilities matrix is provided below.

ROLE	RESPONSIBLE INDIVIDUAL
GOVERNANCE TEAM	
Senior ministry sponsor: Holds overall accountability for the project. Represents the organization that is the major recipient of the system benefits.	
Representatives of relevant Ministry of Health departments: May include planning, finance, and the programs that are involved in the implementation.	
Representatives of donors and technical agencies: Provide guidance, and technical and financial support.	
MANAGEMENT TEAM	
Project manager: Takes on the responsibility for day-to-day direction of the project, communicates with the governance team, and ensures the system is deployed on time and on budget. This role is ideally filled by an influential and skilled person within the Ministry of Health, but can be supported by technical partners. This is the first step toward a successful project. The project manager should possess excellent managerial, technical, and negotiation skills.	
Procurement: Ensures that implementation partners are following organizational guidelines for contracting and licensing.	
Research, monitoring, and evaluation: Individual or team responsible for providing management with the measurement framework and documenting indicators.	
Enterprise architect: Assists with the information technology system architecture and supports the organization's health strategy.	
DEVELOPMENT TEAM	
Business analyst: Models current operations and documents requirements described by the users of the system for the identified health system domain. Identifies areas where improvements in workflow are warranted. Serves as a facilitator who is skilled in methods to reach rough consensus on key decisions and move forward in an efficient manner. This is a key role.	
System analysts, software engineers, and test specialists: Transform business and information requirements into specifications for information systems. Implement specifications using the technologies chosen to support the organization. Validate that the system meets functional and nonfunctional requirements at the proper scale.	
Content, standards, and localization: Create and localize technical documentation, operations manuals, and user manuals.	
DEPLOYMENT TEAM	
Hardware, telecommunication, and networking services: Provide services to procure, deploy, and configure components that are necessary in the field. Provide consumable supplies, such as paper and ink cartridges for printers.	
Training: Train end users and administrators of the system.	
OPERATIONS TEAM	
Data center/hosting: Installs, configures, maintains, and monitors the system within the data center. Applies ongoing security patches and manages data backup and restoration activities.	
System administration: Oversees overall configuration of the health information system. Monitors system use and serves as the secondary level of support escalation from call center operations.	
Help desk/support: Provides different methods of support via email, voice, and in person.	

ANNEX 3. NONFUNCTIONAL REQUIREMENTS CHECKLIST

Use the following checklist as a basis for creating country-specific nonfunctional requirements, as well as for evaluating other systems.

SYSTEM NAME	EVALUATED ON
ORGANIZATION	SUBMITTED BY

CATEGORY: PERFORMANCE

ID	REQUIREMENT	FULLY MEETS	PART. MEETS	PLANNED	N/A
1.1	Make efficient use of data communication time.	□	□	□	□
1.2	Make efficient use of capabilities of lower-cost mobile devices.	□	□	□	□
1.3	Support data capacity considerations (include those for data transmission, storage, and processing) for all users over the expected lifetime of the system.	□	□	□	□
1.4	Use a database that can scale to support projected transaction volume.	□	□	□	□
1.5	Provide real-time response to transactions submitted by connected devices up to the configured national volume level.	□	□	□	□

CATEGORY: COMPATIBILITY

ID	REQUIREMENT	FULLY MEETS	PART. MEETS	PLANNED	N/A
2.1	Use open standards to promote interoperability.	□	□	□	□
2.2	Exchange actionable data between systems (need to enforce semantic interoperability).	□	□	□	□
2.3	Provide access from Internet-enabled devices.	□	□	□	□
2.4	Support flexible models for data collection (e.g., including paper forms, web forms, short message service [SMS] text messages), barcode, etc.).	□	□	□	□
2.5	Comply with industry standards for data exchange.	□	□	□	□
2.6	Interface to open-source or third-party reporting tools.	□	□	□	□
2.7	Comply with industry standards for track and trace of supplies.	□	□	□	□

CATEGORY: USABILITY

ID	REQUIREMENT	FULLY MEETS	PART. MEETS	PLANNED	N/A
3.1	Allow for flexible configurations based on the context of use, including the physical and social environment.	□	□	□	□
3.2	Transmit information in a language (script or voice) that is understood by the user population.	□	□	□	□
3.3	Emphasize ease of use and learnability to reduce training costs.	□	□	□	□
3.4	Be able to be learned by end users and supervisors to meet specified goals of system effectiveness and efficiency.	□	□	□	□
3.5	Enable easy data collection, organization, and dissemination.	□	□	□	□
3.6	Focus on the mobile user experience with secondary use of a computer.	□	□	□	□
3.7	Allow users to find features in two clicks or less.	□	□	□	□
3.8	Enable pleasing and satisfying interaction for the user.	□	□	□	□
3.9	Provide a search interface to reduce data entry burden and improve accuracy on mobile devices.	□	□	□	□
3.10	Support real-time data entry validation and feedback to prevent data entry errors from being recorded.	□	□	□	□
3.11	Support ability to calculate values on behalf of user (eliminating need to add, subtract, multiply, or divide).	□	□	□	□

ANNEX 3. NONFUNCTIONAL REQUIREMENTS CHECKLIST (CONTINUED)

CATEGORY: RELIABILITY

ID	REQUIREMENT	FULLY MEETS	PART. MEETS	PLANNED	N/A
4.1	Enable a task to be canceled and rolled back to previous state.	❐	❐	❐	❐
4.2	Enable users to work offline and then synchronize data when data connection is available.	❐	❐	❐	❐
4.3	Allow a task to be interrupted and resumed.	❐	❐	❐	❐
4.4	Enable earlier versions of a record to be recoverable.	❐	❐	❐	❐
4.5	Enable backup of data so that information is recoverable in the event of a system or hardware failure.	❐	❐	❐	❐
4.6	Accommodate loss of connectivity to hosted application (network may become unavailable while a user is in the process of submitting a form).	❐	❐	❐	❐
4.7	Be able to reliably perform tasks within appropriate time with resistance to failures or deadlocks.	❐	❐	❐	❐
4.8	Be deployable in an environment subject to power loss.	❐	❐	❐	❐
4.9	Allow for client devices with low bandwidth or irregular connectivity.	❐	❐	❐	❐

CATEGORY: SECURITY

ID	REQUIREMENT	FULLY MEETS	PART. MEETS	PLANNED	N/A
5.1	Prevent unauthorized access to patients' protected health information.	❐	❐	❐	❐
5.2	Prevent updates to the database occurring only partially (atomicity), which can cause greater problems than rejecting an entire submission of a form.	❐	❐	❐	❐
5.3	Trace and record changes to data taken by the system and by users (update/delete/add).	❐	❐	❐	❐
5.4	Allow the administrator to establish access privileges and priorities.	❐	❐	❐	❐
5.5	Support definitions of unlimited roles and assigned levels of access, viewing, entry, editing, and auditing.	❐	❐	❐	❐
5.6	Require each user to authenticate by role before gaining access to the system.	❐	❐	❐	❐
5.7	Provide flexible password control to align to national policy and standard operating procedures.	❐	❐	❐	❐

CATEGORY: MAINTAINABILITY

ID	REQUIREMENT	FULLY MEETS	PART. MEETS	PLANNED	N/A
6.1	Be built using technologies that enable local control, open competition, and transparency of the code.	❐	❐	❐	❐
6.2	Have adequate support resources to ensure scalability and sustainability.	❐	❐	❐	❐
6.3	Promote easier acquisition by supporting a range of devices and form factors.	❐	❐	❐	❐
6.4	Able to access the system at all levels/stores.	❐	❐	❐	❐
6.5	Enable local control of operations.	❐	❐	❐	❐
6.6	Be well documented, including known issues.	❐	❐	❐	❐
6.7	Support repair or upgrade of a component in a running system.	❐	❐	❐	❐
6.8	Provide a unique version number for all future updates and releases.	❐	❐	❐	❐
6.9	Enable the system to detect incompatible versions of software running on different components.	❐	❐	❐	❐
6.10	Enable configuration to any national or subnational administrative structure or number of levels.	❐	❐	❐	❐
6.11	Have a support process that tracks and documents bugs from discovery to resolution.	❐	❐	❐	❐
6.12	Enable access to the central system from all levels of the health system.	❐	❐	❐	❐
6.13	Support changes to organizational alignment of facilities and personnel.	❐	❐	❐	❐
6.14	Include an administrable content management system.	❐	❐	❐	❐

CATEGORY: PORTABILITY

ID	REQUIREMENT	FULLY MEETS	PART. MEETS	PLANNED	N/A
7.1	Be able to provide continuity and access to data throughout changes in infrastructure (e.g., telecommunication, power) at the health post level.	❐	❐	❐	❐
7.2	Support extensibility and/or the ability to accept new services or functionality.	❐	❐	❐	❐

ANNEX 4. SELECTION MATRIX

DESCRIPTION	POINTS AVAILABLE	SYSTEM 1	SYSTEM 2	SYSTEM 3	SYSTEM 4	SYSTEM 5
Requirements How well does the system meet the user needs?						
Scalability Has the system been tested or implemented at the scale necessary?						
Sustainability Can the system be easily maintained and adapted as organizational needs change?						
User fit Does the system fit well within the existing culture, language, and user processes?						
Costs Are the costs of implementation and operations within funding constraints?						
Timeline Can the system be implemented within the expected time frame?						
Licensing and contracting How well does the system fit the procurement guidelines for intellectual property and use of local resources?						
SCORE						

ANNEX 5. GOVERNANCE AND DESIGN PRINCIPLES

PRINCIPLE	DESCRIPTION
Business continuity	Organization operations are maintained in spite of system interruptions.
Common use applications	Development of applications used across the organization is preferred over the development of similar or duplicative applications that are only provided to a particular department or unit of the organization.
Common vocabulary and data definitions	Data are defined consistently throughout the organization, and the definitions are understandable and available to all users.
Control technical diversity	Technological diversity is controlled to minimize the nontrivial cost of maintaining expertise in and connectivity between multiple processing environments.
Data are accessible	Data are accessible for users to perform their functions.
Data are an asset	Data are an asset that has value to the organization and is managed accordingly.
Data are shared	Users have access to the data necessary to perform their duties; therefore, data are shared across all functions, departments, and units of the organization.
Data quality	A health information system must have clear rules and methods for handling missing or erroneous data and indicators. Correction algorithms should be developed.
Data security	Data are protected from unauthorized use and disclosure.
Ease of use	Applications are easy to use. The underlying technology is transparent to users, so they can concentrate on tasks at hand.
Flexible and adaptable	A health information system must be flexible in order to adapt itself to changes of all kinds, such as evolving sociologic and economic conditions, changes in the epidemiological situation and the state of health of the population, scientific progress in public health and medicine, and changes in information technology.
Focus on logic of the system before solutions	Analyze and make apparent the logical structure of the health information system first, and then think about computerization. All software within the system, including that for hospital information systems, needs to be developed and applied in a coherent and coordinated fashion.
Make the system visible and easy to understand	A health information system needs to have a logical and transparent structure.
Primacy of principles	These principles of health information systems apply to all parts of the organization.
Reduce burden on data collectors	A health institution sends a report on paper only to the higher-level institution that needs it most or most urgently. It is then up to the latter to distribute it horizontally to those who require it. A higher-level office never requires "summary" reports from the lower level.
Requirements-based change	Only in response to business needs are changes made to applications and technology.
Responsive change management	Changes to the organization's information environment are implemented in a timely manner.
Simplify registers	In a given health institution, there should be only one register for each target population.
Simplify user experience	Coordinate registers and reporting forms by their layout and by a clear designation of corresponding variables. Calculate indicators as much as possible as part of the daily work routine.
Technology independence	Applications are independent of specific technology choices and therefore can operate on a variety of technology platforms.
Use routine data whenever possible	Explore all possibilities of exploiting a health information system for conducting, or facilitating, research studies. Treat sample surveys and related studies as components of a general health information system.

ANNEX 6. PROPOSAL SCORING MATRIX FOR THE SELECTION OF A SOFTWARE DEVELOPER

		COMPANIES				
	MAX SCORE	1	2	3	4	5
PROPOSED SOLUTION						
The proposal successfully communicates a good understanding of the project goals.	15					
The proposal communicates an understanding of the project structure and localities.	15					
The proposed technical solution seems feasible.	25					
The proposed technical solution is complete, addressing all the requirements outlined in the request for proposal.	25					
The proposal presents a logical system of when and where different modes of communication technology will be used (e.g., general packet radio service versus short message service [text messages] versus wired Internet).	10					
The company proactively identified technical or functional challenges and proposed solutions.	10					
TOTALS	100					
PROPOSED METHODOLOGY, WORK PLAN, TIMELINE						
The proposed work plan fits roughly within the timeline.	20					
The proposed work plan includes ample time for project requirements definition and design.	10					
The proposed methodology reflects an approach of high communication with the business owner and users.	20					
The proposed work plan includes time for testing/feedback/iterative development.	10					
The proposal includes all of the key deliverables listed as required in the request for proposal.	10					
The vendor proactively identified issues and risks to the proposed work plan/timeline and offered solutions to mitigate them.	20					
The proposal reflects the requirements stated in the request for proposal related to warranty, pilot support, and maintenance contracts.	10					
TOTALS	100					
ORGANIZATIONAL CAPABILITIES						
Programming and computing capabilities with required architecture, languages, and tools is evident (Python, .NET, C/C#, Java, SQL).	20					
The proposal refers to previous work that is relevant to our project methods and goals.	15					
Sufficient staff are proposed who have appropriate skills and experience.	20					
The proposal contains guarantees for documentation, maintenance, warranty, and ownership transfer.	15					
The quality of the written proposal indicates the company's ability to document appropriately and communicate in the language of interest.	20					
The proposal provides some evidence that the company has experience working with organizations similar to our own.	10					
TOTALS	100					
GRAND TOTALS	300					
PROPOSED PROJECT COST						
The proposed terms and conditions are reasonable and acceptable under contracting policies.						
OTHER						
Level of effort—total work identified in each company's proposal (in days).						

ANNEX 7. VENDOR QUESTIONS

QUESTION	REASONING
What is your largest implementation? How many users? How many records in the database?	Determine if the vendor has experience or evidence that they are able to support the size of your desired implementation.
How many users can use the system at the same time?	If your users typically access the system and provide all of their reports on Friday afternoons, you do not want the system to fail or have very poor performance during those times.
What components of the proposed platform are proprietary? What components use commercial off-the-shelf software? What components are open source?	To follow a principle such as technology independence, knowing the licensing requirements early is important. For system maintenance, knowing the underlying technology and corresponding robustness of either the software provider or the open-source community can be important.
What service-level agreement for uptime do you guarantee each month? How many hours of maintenance is the system unavailable each month and when are those typically scheduled?	What amount of time is tolerable for the system to be unavailable? A total of 95 percent uptime translates to eight hours each week. Usually, the vendor will apply security updates to the software on your behalf. Yet, you would not want this to occur during key periods of system use.
How would you integrate with our health information system? Can you provide examples of how you have done this before?	If an integrated system is a key principle, knowing that the application has a demonstrated architecture for data exchange is necessary. If the integration has never been done before, it may be considered an unsupported customization that requires ongoing maintenance fees.
How do you safeguard the security and privacy of our data? What were the results of your most recent external audit?	Data security is critical for health information systems. Information such as patient records can be hard to replace and should not fall into the wrong hands. Data can be lost due to disasters such as a flood or fire but also to hackers or a malware infection.
How often would our data be backed up? Can you provide your disaster recovery plans? When was the last exercise and what were the results?	If data are an asset, knowing that the vendor has processes to store and restore your system in the event of an emergency is important.
What training and support services do you provide? What hours is support available?	Early clarification of roles and responsibilities for deploying the software is needed to understand the overall costs. Training users is often a large part of the deployment budget. Sometimes, the vendor will provide training for administrators and train your trainers. Do your normal hours of operation coincide with the support hours provided?
What languages does your application support?	For ease of use, the system user interface should be in the language of your users. If the language is not currently supported, ideally the vendor has capabilities that allow you to localize the various terms.
What is the annual maintenance and licensing fee? How much is this expected to increase annually?	Sometimes hidden fees obscure the true costs of the system. Maintenance fees of upwards of 20 percent of the software license may be required when the contract is signed.

ANNEX 8. WHAT DRIVES COSTS IN ALL PHASES OF THE PROJECT LIFECYCLE?

CATEGORY	COST DRIVERS
GOVERNANCE	• Number of trips and meetings.
MANAGEMENT	• Head count, salary, and travel expenses. • Mix of local versus international technical assistance.
DEVELOPMENT	
Software and interfaces: Configuring the environment, customizing modules, and developing interfaces to other parts of the health system take time and money in addition to any fees incurred for software licensing.	• Number of user requirements or stories to be developed. • Licensing costs per environment (production, test, training). • Licensing costs per user. • Number of interfaces required and level of effort.
Content, standards, and localization: If the system is not already available in the local language, some costs may be incurred to modify documentation and the user interface.	• Number of additional languages not currently supported.
DEPLOYMENT	
Client hardware: Includes components such as computers, printers, and scanners. If the system will be using mobile devices, there is a need to determine if existing hardware can be used, or if new mobile devices will need to be purchased.	• Number of users on desktop computers and mobile devices. • Cost and availability of data connectivity and power.
Training: The cost of developing and delivering training to the staff in the appropriate language. Includes ongoing training for software updates and to address the needs of new users. Includes the per diems for training participants, transportation costs of bringing individuals to the training events, and facility fees.	• Number of users to be trained. • Days of training per user. • Days of refresher training per year.
OPERATIONS	
Data and communication services: Voice and data services to support the data and communication flow.	• Internet connectivity. • Monthly mobile data plan. • Expected minutes and data use per user. • Number of text messages.
Hardware maintenance and replacement: Desktop computers need to be upgraded and maintained; lost or stolen mobile devices will need to be replaced; etc.	• Number of hardware devices. • Replacement rate. • Maintenance costs per device.
Server management and hosting: The cost of internal and external support needed to support the system for both software and hardware problems and maintenance.	• Data center setup. • Currently supported hardware and software infrastructure. • Service levels. • Software and hardware maintenance fees.
Administration and call center support: In order for local individuals to manage day-to-day operations, there is often an administrator training that allows a small group to configure and modify the system.	• Expected percentage of support calls. • Call center staffing hours. • Additional support staff required at the national and subnational level. • Equipment replacement percentage. • Staff turnover.

ANNEX 9. TOTAL COST OF OWNERSHIP BUDGET MATRIX

The table below can be used to summarize costs across categories.

BUDGETING CATEGORY	YEAR 0 (INITIAL LAUNCH)	YEAR 1	YEAR 2	YEAR 3
Governance				
Meetings and administrative support				
Management				
Overall project management				
Research, monitoring, and evaluation				
Development				
Software and interfaces				
Content, standards, and localization				
Deployment				
Client hardware				
Training				
Operations				
Data and communication services				
Server management and hosting				
Administration and call center support				
TOTAL				

Note: Ideally, costs of international support decrease over time with a corresponding increase in local health systems strengthening.

ANNEX 10. WORK PLAN

#	ACTIVITY	MILESTONES								
		1	2	3	4	5	6	7	8	9
1	PLANNING									
1.1	Define outcomes and form vision	X								
1.2	Negotiate scope of project	X								
1.3	Establish project charter and governance	X								
	Milestone review	X								
1.4	Form initial team		X							
1.5	Define requirements and expected results through user involvement		X							
1.6	Find the right solution—seek evidence and lessons learned from similar projects		X							
1.7	Establish budget		X							
1.8	Develop draft implementation plan		X							
	Milestone review		X							
1.9	Post additional positions			X						
1.10	Hire and/or obtain short-term technical assistance			X						
1.11	Select the right vendor(s)			X						
1.12	Negotiate/finalize vendor contract(s)			X						
	Milestone review			X						
1.13	Finalize implementation plan			X						
1.14	Finalize monitoring and evaluation plan			X						
1.15	Draft communication plan			X						
	Milestone review			X						
2	MANAGEMENT AND COMMUNICATIONS									
	Schedule the following items…									
2.1	Project manager progress reports		X	X	X	X	X	X	X	X
2.2	Working group meetings		X	X	X	X	X	X	X	X
2.3	Updates for senior management	X	X	X	X	X	X	X	X	X
2.4	Governance committee meetings	X	X	X	X	X	X	X	X	X
2.5	Communication to organization			X	X	X	X	X	X	X
3	DEVELOPMENT									
3.1	Hold project kickoff meeting				X					
3.2	Establish risk and change management plan				X					
3.3	Perform gap analysis				X					
3.4	Finalize technical protocols, standards, and operating systems				X					
3.5	Redesign processes and tools for optimization				X					
3.6	Design training programs				X					
3.7	Obtain approval for new processes, tools, and training plan				X					
3.8	Finalize training strategy and plan				X					
	Milestone review				X					

#	ACTIVITY	MILESTONES								
		1	2	3	4	5	6	7	8	9
3	**DEVELOPMENT (CONTINUED)**									
3.9	Complete capacity sizing and needs assessment					X				
3.10	Gather accurate performance data for existing systems and processes					X				
3.11	Establish data center and hosting environment as needed					X				
3.12	Install and configure server environment					X				
3.13	Test integrations					X				
3.14	Prepare user acceptance testing scripts					X				
	Milestone review					X				
3.15	Set up call center and train support staff						X			
3.16	Perform user acceptance testing						X			
3.17	Resolve high- and medium-level issues						X			
	Milestone review and signoff on phase						X			
4	**DEPLOYMENT**									
4.1	Procure items (might be staggered based on rollout strategy)						X			
4.2	Print updated tools						X			
	Milestone review						X			
4.3	Execute communication plan at subnational level, districts, and health facilities						X			
4.4	**PILOT**									
4.4.1	Prepare implementation checklist							X		
4.4.2	Install network infrastructure (telecommunication and power)							X		
4.4.3	Install hardware and software at pilot sites							X		
4.4.4	Provide updated tools, as necessary							X		
4.4.5	Pilot test processes, tools, and technology							X		
4.4.6	Resolve high- and medium-level issues, modify configuration as necessary							X		
4.4.7	Revise implementation checklist and training							X		
	Milestone review							X		
4.5	**ROLLOUT**									
4.5.1	Install hardware and software								X	
4.5.2	Deploy new tools								X	
4.5.3	Execute training for revised processes, tools, and technology								X	
	Milestone review								X	
	(Repeat above for each rollout cycle)							X		
5	**OPERATIONS**									
5.1	Implement monitoring process and tools									X
5.2	Finalize service-level agreements and maintenance contracts									X
5.3	Establish backup procedures									X
5.4	Monitor use and maintenance needs									X
5.5	Evaluate system performance									X
	Milestone review									X