

Field Guide

- [Assessing Needs](#)
- [Project Design](#)
 - [Collaborating with government and NGOs](#)
 - [Connecting multiple sites](#)
 - [Entering data at point of care vs. retroactively](#)
 - [Using a vertical program model vs. primary/comprehensive care model](#)
- [Infrastructure Requirements](#)
 - [Power infrastructure](#)
 - [Connectivity](#)
 - [Security](#)
- [Machines for entering, storing, and accessing data](#)
 - [Server requirements](#)
 - [Work stations for point of care](#)
 - [Mobile devices](#)
- [Installing OpenMRS](#)
- [Constructing Forms](#)
- [Staff Requirements](#)
 - [Primary implementer](#)
 - [Long-term IT support](#)
 - [Data entry clerks](#)
 - [Data managers](#)
- [System maintenance and performance](#)
 - [Common issues and solutions](#)
- [Reference Implementations](#)
- [Additional Resources](#)
- [Field Guide Contributors](#)

The OpenMRS Field Guide is a new and incomplete resource that aims to help implementers with the nuts and bolts of beginning and sustaining a successful implementation of OpenMRS, from securing electricity to maintaining data quality.

This page aims to complement existing resources such as the [Implementation Overview](#) and [PIH model online](#), highlighting details from actual experiences on the ground. Many of the topics in the Field Guide are not about OpenMRS per se, but about the many components of infrastructure and staff practices that must be in place for OpenMRS to function. If you find similar or complementary resources, please add them to the list at the bottom of this page.

If you don't find what you're looking for in this general guide, please ask the [Implementer's mailing list](#) (subscribe on the [community](#) page). The list is full of people just like you who would be happy to answer your questions promptly.

There is also an OpenMRS Book for Implementers available: <http://booki.flossmanuals.net/openmrs-guide>

Assessing Needs

Get the stories and work flows of:

- Patients
- Providers (doctors, nurses, clinical officers)
- Data clerks/record keepers
- Lab and pharmacy staff

Find out how all medical information is supposed to be managed as well as how it is actually managed. Note that practices may vary seasonally, for example if the hospital is much busier due to increased malaria during rainy season or malnutrition before harvest.

People generally want to be positive in describing their work places, so you may need to ask multiple people multiple times. Get copies or pictures of all paper forms if possible. Figure out where (i.e. specific rooms and desks) data is recorded onto paper and by whom. Write an overview of current practices and define specific shortcomings that could be addressed by using an electronic medical records system.

Project Design

The OpenMRS platform is flexible enough to support a wide variety of use cases or implementation models. To adopt and adapt the model that best fits your project, you will need to broadly consider assessed needs and available resources, including access to basic infrastructure, hardware, community partners, and expertise of staff, volunteers, and available contractors. You may find it helpful to perform a [SWOT analysis](#) of your project. Major issues to consider include:

- [Collaborating with government and NGOs](#)
- [Connecting multiple sites](#)
- [Entering data at point of care vs. retroactively](#)
- [Using a vertical program model vs. primary/comprehensive care model](#)

Collaborating with government and NGOs

- You need to know the reporting and regulatory requirements for the Ministry of Health in your country. If you are working at a public clinic or hospital, it is important to learn those before completing the design of the model. For multiple sites, begin talking with the Ministry of Health as early as possible to consider how the model could be replicated.

- If there are NGOs in the area that already use OpenMRS or plan to use it, consider the possibilities for sharing a system of unique national IDs and/or sharing a concept dictionary so that medical information can be more easily transferred from one site to the next.
- You can begin to reach out to other NGOs by asking the implementers mailing list ([Template:Email:implementers](#)) if there are any other implementers in your region. Whether or not you end up collaborating with them, it can be very helpful to communicate with a local experienced implementer during the design phase of your project.

Connecting multiple sites

If you have more than one site using OpenMRS, you will need to identify an effective way to connect them. Possible options include:

- A central server with remote sites connected via Internet
- Local servers connected to work stations with a wireless local area network
- Local server connected to work stations by Ethernet cable
- A collection of sites, each with its own server, are connected by Remote FormEntry module or sync feature
- Paper forms are completed at remote sites and brought to a central location where they are entered into a local server

Entering data at point of care vs. retroactively

See [Point of Care](#) page.

Using a vertical program model vs. primary/comprehensive care model

In the future it would be great to collect brief descriptions of diverse implementation models that could use a case study format to highlight how past projects have navigated these project design issues. Maybe on the [Reference_implementations](#) page.

Infrastructure Requirements

Existing infrastructure is a critical factor in designing your project. Implementing an electronic medical records system can be a good motivation to build infrastructure such as stable electricity or Internet connectivity that has tangential benefits for other aspects of the health facility. However, it is critical that project goals are modest enough that they will be successfully implemented, embraced by local staff, and have sufficient long-term financial support.

Power infrastructure

OpenMRS is only as reliable as the power system that supports it. Unless electricity is almost 100% stable in your area, you will probably want a primary power source and a backup.

- If you are building a new electrical system, use a skilled electrician -- Poorly designed systems are unreliable and can cost more in the long run. For example, systems that allow batteries to discharge completely will dramatically shorten their lifespan. Worse, improperly designed systems can be dangerous, especially with the high currents that DC systems generate. Batteries should be properly protected and vented. If you hire someone to install your power system, check that they have proven competence in both computing and power technologies.
- Think low maintenance -- In off-grid settings, low maintenance often means solar, as fuel and maintenance costs for generators quickly offset upfront savings and are a major failure point for projects. Batteries should be maintenance-free, sealed lead-acid gel (SLA), preferably the AGM reinforced type. These are somewhat more expensive than the "wet" or "flooded" cell batteries that require regular maintenance and checkups, but are much more reliable in the long term.
- Consider hybrid systems -- Hybrid systems allow batteries to be charged by more than one power source; for example, both solar panels and a generator, or generator and intermittent grid power. Although hybrid systems are slightly more complicated and expensive to deploy, the redundancy makes them more reliable. Reliability of your power source will be particularly important if you are considering implementing a [Point of Care](#) system.
- Include surge protection and voltage stabilization -- Generator or partial-grid power can easily damage computers and networking equipment. Spikes in power or low voltage are the most common cause. The equipment to protect against this danger is relatively inexpensive and readily available in most places and offers cheap insurance against damage to more costly equipment. These components should always be used with an AC power source.
- Size systems carefully and conservatively -- Where power is scarce, it is valuable. If you allow staffs to use your power source for charging personal cell phones, DVD players etc, they will consider isolating ICT power systems from others. If your server relies on a battery backup, make sure the battery powers only the OpenMRS system.
- Build around locally available inputs -- Imported parts that are not available locally tend not to be replaced if they unexpectedly stop working. Develop relationships with several suppliers of high-quality batteries, chargers, solar panels etc.

Organizations that have helped OpenMRS implementers set up power systems:

- Inveneo has worked with various OpenMRS implementers. They have a network of [certified ICT partners](#) who can provide tools locally throughout much of Africa
- [SELF](#) has helped install solar panels with diesel generator backup to power a Partners in Health clinic in Rwanda and a Village Health Works clinic in Burundi
- [Catapult Design](#) has helped health clinics in Rwanda install solar panels

Connectivity

How will you connect your work stations to your server?

- Remote servers connected via Internet. Work stations connected to Internet via wireless router or Ethernet cable
- Local server connected to work stations with a wireless local area network
- Local server connected to work stations by Ethernet cable

Security

Some projects require significant investment to protect ICT and power infrastructure while others do not. Considering the context, only local knowledge can guide this decision.

- Use "passive" security as much as possible (e.g. bars versus locking shutters)
- Have metal doors with locks, and bars on all windows
- Use a large, sturdy, lockable metal cabinet to store the server, fans, and backup batteries.
- Avoid glass especially at ground level
- Having a relatively public location can help minimize theft but only if solid basic protections are in place
- Secure solar arrays with locking frames. Based on local conditions, wire mesh or other screening may be required to protect glass from rocks, etc. If available, use impact resistant panels.
- Widespread community support for the ICT system and/or the services they support can help to minimize the likelihood of theft or vandalism.
- A laptop may be more likely to be stolen because it is portable and can be used for purposes other than the EMR. A desktop computer is less portable, and a thin client such as a basic touch-screen device may be less vulnerable to theft if it can't act as a standalone computer - i.e. can't function when it isn't connected to a central server. Using a thin client has the added benefit of ensuring that staff will use the device for work purposes rather than browsing the web, making Word documents etc.

Machines for entering, storing, and accessing data

Once you have a good understanding of the kind of system you want to set up, the amount of data you want to collect over time, and the type of system your infrastructure (either pre-existing or built specifically for this project) can support, you can select server and work stations that meet your needs.

Server requirements

Most installations require a minimum of 2GB RAM and 250GB hard disk.

At current installations:

MVP Uganda: Every day six people at two sites enter a total of 80-100 forms with 20 obs per encounter per person per day. They have two sites accessing an Inveneo server with 1 GB of ram. After about one year with 40K encounters and 20K patients the server started to run slowly.

AMPATH Kenya (as of June, 2010): 230K patients, 70 million obs. Poweredge R510 Two Processor,E5540, 2.53/5.86, 8MB Intel Dual Socket Nehalem, 8 Memory modules 4GB 1333 MHZ. 40+ people accessing the system at a time.

HAS Haiti: 600K patients, 4.4 million obs. Average entry of 150 forms and 1400 obs per day. 12GB memory, Dual-Quad Core Intel Xeon 2.33GHz server. Up to 11 users accessing the system simultaneously.

Work stations for retrospective data entry

If you intend for clinicians to enter data on paper forms that are later entered into OpenMRS by data entry clerks, you should estimate the number of work stations you will need based on the amount of data you will enter. As a general rule, it will take a data clerk about one day to enter 80-100 forms with 20 observations per form (this general rule may vary greatly from site to site). For retrospective data entry you may use regular computers that range in cost from \$300-\$1,000.

If you intend to use InfoPath (for a FormEntry module or Remote FormEntry module), you will need to use the Windows operating system (minimum of Windows XP).

Work stations for point of care

If you intend to build a point of care system, you should estimate the number of work stations needed based on the number of clinicians or clinic rooms. You might want to consider using thin clients, touch-screen devices or mobile devices if for remote clinics. Such devices range from \$100 (for a J2ME smartphone capable of running OpenMRS) to \$1,000 for some touch-screen appliances.

Mobile devices

If you would like to submit forms from the field and you do not need rich access to patient data, you can do so with mobile devices that cost as little as \$20. Forms can be sent via SMS, GPRS, or WiFi. On the other, end of the spectrum are applications that run on more expensive smartphones with features that are nearly equivalent to a regular OpenMRS workstation, except on a smaller screen.

Read about the various mobile tools that integrate with OpenMRS on the [Mobile](#) page.

Installing OpenMRS

See [Step by step installation for implementers](#).

Constructing Forms

Building forms is an iterative and non-trivial process. Consider various options for creating and submitting forms, including HTML form entry module, XForms module, FormEntry module (uses InfoPath) and Remote FormEntry module (also uses InfoPath).

Staff Requirements

Primary implementer

- One dedicated IT person for at least a month or two, usually full time. They need to know the whole stack (server maintenance, MySQL, Tomcat, Java, etc.)
- Ideally, this person will work daily during the entire implementation process with the local staff member who will eventually be in charge day-to-day maintenance of the system
- The level of expertise you need on site depends in part on whether or not you have the Internet to get help from online resources or remote developers

Long-term IT support

Once OpenMRS is running smoothly, it is still necessary to have an IT person on site or nearby who can respond to problems with the infrastructure that OpenMRS relies on. Issues may arise with:

- Electricity
- Connectivity to the Internet or a local area network
- Computer viruses, software updating and basic computer problems e.g. a mouse stops working

Many organizations lack the internal skills to deal with these issues, so external support can be critical in general:

- Budget for support and maintenance – Annual maintenance cost is usually 5% to 15% of the original total budget but costs will vary significantly based on terms of contract.
- Avoiding problems is easier and less expensive than fixing them -- For example, to keep solar power systems working properly, it is important to clean panels regularly.
- Teach users about proper equipment care – New computer users may inadvertently mistreat equipment. User training should include basics about how to keep equipment in good working condition (e.g. to be careful not to spill food or drink on keyboards, not to cut power to equipment while it is running, etc.) The overall goal should be to make users feel a degree of ownership of the equipment.
- Define a tiered support strategy – Train on-staff system administrator(s) on basic maintenance and troubleshooting and give them a direct link to external expert support. Simple troubleshooting cheat sheets can help in settings where system administrator turnover is high.
- Purchase pro-active site visits – These visits are especially useful during the first 3-6 months, when misuse or equipment failure is most likely. Support providers can often be enlisted to provide training, as needed, for system users.
- Buy support from whoever installed the system – It can be hard to identify the source of ICT problems, much less to assign responsibility for fixing them. Purchasing support from the equipment provider helps to avoid this challenge. Make sure you negotiate for both up front.

Data entry clerks

Many implementations use existing clerks though some train cleaners to enter data.

Data managers

If you have more than 4 data entry clerks, it is often useful to train data managers to oversee the data entry clerks and ensure that they are maintaining high data quality.

System maintenance and performance

See [Step-by-Step_Performance_Tuning](#)

For other bugs/issues, you might be best of contacting the [Template:Email:implementers](#) mailing list.

Common issues and solutions

Issue: The system slows down every time I try to run a report, making it impossible for the data clerks to enter data.

Solution: Many large sites will replicate their database and generate reports on the replicated database while continuing to enter data on the primary database.

Issue: We frequently have to reboot our system due to Tomcat out of memory errors.

Solution: The 1.5 release of OpenMRS fixed a few memory leaks, so upgrading might solve your problem. You can download the release on the [Downloads](#) page.

Issue: Running multiple instances of OpenMRS on a single server slows things way down.

Solution: If you are currently running multiple servers to support multiple instances of OpenMRS, you might want to create a cluster and virtualize each instance of OpenMRS so that they can run on the same cluster.

Reference Implementations

Eventually this page will document reference implementations using a case study approach aimed at sharing practical experiences in diverse settings with new OpenMRS implementers.

Additional Resources

- [EMR section of Partners In Health Model Online](#)
- [Inveneo Primer on ICT Sustainability](#): This doc was salvaged with permission and re-purposed as the basis for much of this Field Guide's section on dealing with hardware and infrastructure
- [HackMySQL](#): Simple and to the point documents on dealing with MySQL

Field Guide Contributors

Isaac Holeman of [FrontlineSMS:Medic](#)