Modulo Basico Phase 1 Initial Assessment and Recommendations for the Future

April, 2009

Activities during:
Oct 2008 – Jan 2009

Bill Lober, MD
Associate Professor
Christina Quiles
Technical Program Manager

Contact information:

Dr. Bill Lober
MS 357266 - BNHS
1959 NE Pacific St
Seattle, WA 98195 USA
+1 206 616 6685 (voice)
+1 206 222 2455 (fax)
lober@u.washington.edu

Summary of Activities and Recommendations

This summary addresses both of the scope of this assessment, and the primary options and highest priority recommendations that came out of that assessment. A wider range of specific issues and recommendations are detailed in the report, as well as a more thorough discussion of the priority recommendations.

Overall, Modulo Basico has demonstrated initial success in supporting routine data entry, facilitating data transfer and storage, and providing basic multi-level reporting. The staff supporting the development and deployment of this system should be commended for their effort and implementation success. We feel that it is important to continue with Modulo Basico, focusing efforts on evolving the current model to adhere with the Ministry's vision for the future of aggregate reporting and case surveillance in Mozambique, as well as to increase interoperability with emerging international standards for data management and transmission.

Scope of Activity

In October 2008, the International Training and Education Center on HIV (I-TECH) was engaged to conduct an assessment of the Modulo Basico system, which is developed, maintained, and used by the Mozambique Ministry of Health (MOH) to provide multi-level, national monitoring and evaluation of a diverse set of health care programs.

The assessment work plan is included as Appendix 1, and describes the two Phases of this project. Phase 1 consists of an initial assessment aimed at documenting key aspects of the system and assessing its current state and stability. Phase 2 focuses on the development of more detailed system and process requirements. Specifically, the Phase 1 goals which are addressed in this report, are:

1. Document process and type of information flows handled by current Modulo Basico (MB), including who and how people interact with the current paper-based and electronic information flows.
2. Document key aspects of Modulo Basico database architecture.
3. Identify conditions, resources, and inputs required to sustain Modulo Basico stability in coming 1-2 years, including human resource capacity needed for system maintenance.
4. List types and scope of additional system documentation necessary, given scenario of continued use of existing Modulo Basico in coming 1-2 years.
5. Identify performance gaps, including gaps in data quality or timeliness, associated with current Modulo Basico.
6. Describe recommendations for updated, enhanced system functionality to meet MOH's needs.
7. Outline possible system architecture and associated IT platforms for redevelopment of Modulo Basico to achieve these recommendations, with general cost and risk estimates.
8. Outline 2-3 scenarios for scope of redevelopment of Modulo Basico (which may ultimately be undertaken in a phased manner), with general cost, time, and risk estimates.

Three Basic Options

There are three distinct alternatives for the development or evolution of Modulo Basico:

1. The current application could be maintained while documentation and training were enhanced. However, concerns with this approach include instability of the current development platform and difficulty in adding new features.

2. The current application could be replaced with an existing open-source or commercial software product. Functionality would be improved immediately; however, this approach does not take advantage of the significant existing implementation of Modulo Basico, nor of the features of the system that work well. Furthermore, this approach has all of the risks of any major software implementation project.
3. We recommend an incremental approach to both improving and replacing Modulo Basico, which combines elements of each of these approaches. We recommend some modest stabilization and development of interoperability features for the current application, combined with the acquisition or development of a more feature rich data warehouse application, using the same interoperability standards. This would be implemented initially at the Ministry level, and then, driven by experience, need, and availability of resources, expanded to replace the stabilized version of the present software at first the provincial and eventually the district level.

Goals and Principles

We believe the primary goals of Modulo Basico development should be to ensure a system is in place which meets the core requirements of data collection, quality management, and data usage, and which is aligned with the future, and perhaps evolving, vision of MOH. The most important secondary goals, in our opinion, include flexibility in the development plan, local capacity development and incremental assumption of full responsibility for the system, clear documentation and best practices in development of the system, and good training to ensure that the system is used efficiently, that the data are accurate, complete, and timely, and that the users are able to ensure that the information in the system can find its way out – that the system is useful and valued.

In order to ensure the long term viability of the Modulo Basico system, we believe it is important in the short term to maintain existing program relationships and strengths, such as the national network of staff involved with data collection and data quality at the district and provincial levels, and the collaboration between specific programs at the Ministry. It is important that data continue to be reported through Modulo Basico to maintain consistency and because it is more complex and expensive to implement a new system than it is to evolve a working system. Furthermore, the system should be actively supported and developed to ensure it continues to meet programmatic needs.

Priority Recommendations

At the risk of oversimplifying these recommendations, we believe the following are important short term goals:

1. Maintain the existing reporting and data management activities using Modulo Basico.
2. Migrate Modulo Basico to a new development environment compatible with the newer Windows operating systems and capable of cross-platform deployment on Linux.
3. Improve user's guide documentation
4. Redesign the database architecture so it better supports both reliability and future extensions of the system.
5. Improve data transmission to incorporate both standards-based approaches, and features to enhance reliability and redundant transfer of information.
6. Developed a set of visualizations to make clear both quality and consistency of the data and to provide visual summaries and comparisons of indicators.

7. Identify resources within the Ministry, or through collaboration with a suitable partner, to ensure that these short term goals are met, and that planning proceeds for the long term support and evolution of the system.

We believe the following are important long term goals:

1. Take advantage of a local informatics Group, such as a company or the M-OASIS group at UEM, combined with technical assistance for design and oversight, to provide programming services, with the intent of developing a strong local software development capability for Modulo Basico.
2. Define processes for requirements definition and modification, development, and deployment that are transparent and well described.
3. Take advantage of existing programmatic expertise, relationships, and strengths, while ensuring the development of informatics capacity in Mozambique to further evolve and maintain the system.
4. Further assess the administrative and support requirements at the district and provincial levels and ensure that there is adequate support for the system.
5. Foster collaborative exchange of information about system use, report development, etc. at the Ministry and provincial levels.
6. Develop both context-sensitive help and training materials, accompanied by a training plan which includes both technical issues such as installation and maintenance, and data use issues, such as appropriate site level analysis.
7. Develop features and visualizations to support improved data completeness and timeliness, along with district and provincial level procedures for data quality review.

Table of Contents

Acronyms

ART	Antiretroviral therapy
ARV	Antiretroviral (medication)
AJAX	Asynchronous JavaScript and XML
BES	Weekly Epidemiological Bulletin (BoletimEpidemiologicoSemanal)
CDC	US Centers for Disease Control and Prevention
EMR	Electronic medical record
GAP	Global AIDS Program (CDC)
I-TECH	International Training and Education Center on HIV
M-OASIS	Mozambique – Open Architectures, Standards, andInformation Systems project
MB	Modulo Basico
MISAU	Ministério da Saúde
MOH	Ministry of Health
OS	operating system
PEPFAR	President's Emergency Plan for AIDS Relief
UCSF	University of California San Francisco
UW	University of Washington
UEM	Universidade Eduardo Mondlane
VB6	Visual Basic 6
WHO	World Health Organization

Technical Glossary

Architecture (System) is the design and functional structure of an information system, including its software components, the software technologies upon which those components depend, the formats used to exchange information, and the protocols or circumstances that cause those information exchanges to occur.

AJAX: stands for Asynchronous JavaScript and XML, and it is a web development technique that allows a web page to access data from the server asynchronously (e.g., a drop down populates from the database in response to a user clicking on the drop down box.)

Client: Client machines are the actual hardware that the users of the system utilize when entering data into MB or another electronic system. Similarly, client software is the software, such as a web browser, used to access an information system

Database architecture: The schema for organizing a collection of stored data within a relational database, including the relationships between fields, records and files.

eHealth(also written e-health) is a relatively recent term for healthcare practice which is supported by electronic processes and communication.[1]

Instance: A single installation of a non-networked software product.

iSANTE: An electronic medical record system currently implemented at over 40 sites in Haiti for HIV patient tracking, case surveillance, and monitoring and evaluation.

Java: A programming language syntactically derived from C and C++ but with a simpler object model. Java is known for its portability, or its ability for software written in Java to be used on different software platforms.

Open source: An approach to software development in which programmers can read, redistribute, and modify the source code for a piece of software, resulting in community development of a shared product.[2]

MySQL: An open source database management system. (www.mysql.org)

National health information system architecture: Another term for this is "national enterprise eHealth architecture." The schema for a national health information system, describing relationships and data flow between component systems. Component systems may involve both paper-based and electronic elements, and may include electronic health records, laboratory information systems, human resources for health information systems, supply chain management systems, demographic information systems, population health information systems, and systems for aggregate reporting of health services data.

OpenMRS: a community-developed, open-source, framework for developing electronic medical record systems. (www.openmrs.org).

PHP: A scripting development language that is used to create dynamic web pages.

---

[1] http://en.wikipedia.org/wiki/EHealth (accessed January 29, 2009)

[2] Shortliffe, EH and Cimino, EJ (Eds.) Biomedical Informatics: Computer Applications in Health Care and Biomedicine, Third Edition. Springer: New York, 2006.

Portable language: A development language which allows programs written in it to be run similarly on multiple supported platforms.

Software platform: The computer operating system and associated programming languages required to run a software application.

System architecture: The schema for the components of a specific health information system, describing how data flows between the components. System components may include paper-based tools, stand-alone instances of software applications, and/or networked software applications.

Web-based application: An application that can be accessed through a web browser over a network. This framework facilitates updates and maintenance to multiple instances of a program, by allowing the upgrade maintenance of multiple running instances at one time.

Web browser: A computer program used to access and display information resources on the World Wide Web.[1]

Web server: A computer program that is responsible for accepting HTTP requests from a browser and formulating HTTP responses, working as somewhat of a translator between the backend (e.g. database, etc.) and the web browser or client display.

Modulo Basico Overview

Since 2003, The Ministry of Health (MOH) has supported the development, use, and enhancement of a system to collect, transfer, and manage routine health information for the purpose of program monitoring and evaluation.

Modulo Basico (MB) is a simple instrument for automating aggregation of data in paper forms that make up the National Health Information System (including its subsystems), supported by trained staff at the district, provincial, and MOH level, which is used to collect and combine summary data on health care services as well as data on the epidemiological profile of patients from individual health care facilities. Modulo Basico uses 15 different paper forms representing specific vertical programs or areas of service (see Appendix 2). Typically, these paper forms are transported to a District Directorate of Health, where they are entered into the software system. Then, both the paper forms and an electronic copy of the entered data are transported to the Provincial Directorate of Health. In some cases, where the data cannot be entered electronically at the district level, the data entry for that district may be done in the provincial office. Data for the entire province are transmitted to MOH Department of Health Information centrally, and stored in a single database. Most reporting is done monthly, though surveillance of epidemic diseases occurs on a weekly basis.

Modulo Basico, in summary, was developed to support routine data entry, facilitate transfer and storage, and provide multi-level reporting. The system was designed and built by a consultant funded by the GTZ[3] organization. The consultant was based within MOH and, until the end of November, 2008, also oversaw the training and implementation activities.

Both a technical description of the information system and a more detailed description of its function are provided in the Findings section.

Phase 1 Assessment – Scope and Methods

In October 2008, the International Training and Education Center on HIV (I-TECH) was engaged to conduct an assessment of Modulo Basico. The focus of this assessment was on the technical characteristics of the system, and to determine the current stability and efficiency of the system itself, rather than the policies and procedures around the collection and use of these data.

The work plan is included as Appendix 1, and describes the two phases of this project. Phase 1 consists of an initial assessment aimed at documenting detailed technical and operational aspects of Modulo Basico and assessing the short and long term requirements necessary to maintain the system as well as enhance its utility. Phase 2 focuses on the development of a refined reporting infrastructure and detailed process requirements, as well as documents to support and facilitate decision making. I-TECH's technical capacity in public health informatics is described further in Appendix 2.

[3] http://www.gtz.de/en/

Phase 1 Objectives

The main goal of Phase 1 was to perform an in depth evaluation aimed at documenting key aspects of Modulo Basico and evaluating its current state and stability. Specifically, we focused on identifying aspects of the system that can be improved to reflect the vision of the Ministry (MOH) and Department of Health Information and to respond to program needs in the short and long term. Concerns the ministry expressed with the current system's ability to ensure system stability include:

- complete documentation of the current system,
- both database and application design,
- long term strategic planning,
- data quality, and
- adequate training and information transfer regarding the development and implementation of Modulo Basico.

Based on these priorities, we focused on the following core objectives:

1. Document process and type of information flows handled by current Modulo Basico (MB), including who and how people interact with the current paper-based and electronic information flows.
2. Document key aspects of Modulo Basico database architecture.
3. Identify conditions, resources, and inputs required to sustain Modulo Basico stability in coming 1-2 years, including human resource capacity needed for system maintenance.
4. List types and scope of additional system documentation necessary, given scenario of continued use of existing Modulo Basico in coming 1-2 years.
5. Identify performance gaps, including gaps in data quality or timeliness, associated with current Modulo Basico.
6. Describe recommendations for updated, enhanced system functionality to meet MOH's needs.
7. Outline possible system architecture and associated IT platforms for redevelopment of Modulo Basico to achieve these recommendations, with general cost and risk estimates.

Additional Goals for Phase 1

In addition to the stated objectives of the work, we believe there are several other important goals to be considered as part of the initial assessment of Modulo Basico, though these may be more directly addressed during Phase 2:

1. Identify changes to MB system architecture so that it can fit with initiatives to evolve a common national enterprise eHeath architecture, such as those presently being sponsored by the WHO Health Metrics Network and the Rockefeller Foundation.

2. Identify changes to MB to acknowledge developing international standards, such as Indicator Exchange Format (IXF) for transmission of aggregate data.

3. Identify changes to workflow and software functionality to support improved data quality.

4. Identify changes to workflow and software functionality to support interpretation of data through visualization and access by analysis software.

5. Identify opportunities for broader use of data at the district and provincial levels.

Information Gathering Activities

This section briefly summarizes, in chronological order, the activities we undertook to learn about Modulo Basico and its use. Specific findings are grouped according to topic and presented in more detail in the Findings section. Appendix 4 includes a more detailed list of the topics and themes of each information-gathering activity that took place during our October, 2008 visit to Mozambique.

Activities in Mozambique

The initial assessment visit to Mozambique took place from October 20-27, 2008. The visit team included Dr. Bill Lober (I-TECH/UW), Ms. Christina Quiles (I-TECH/UW) and Dr. Janise Richards (CDC Global AIDS Program). In Mozambique, we interviewed stakeholders and users within the MOH at the health center, district, provincial, and national levels in order to gain an understanding of Modulo Basico, and of the various ways in which it is used. We also met with representatives of NGO's and University affiliates to discuss their experiences and impressions of the system. We reviewed the development, maintenance, and use of Modulo Basico at the national level, including a thorough introduction to the Modulo Basico software from the primary developer. In these meetings, we discuss how the application meets monitoring and evaluation needs, opportunities for improvement in the system, the current capacity for healthcare IT activities in Mozambique, as well as potential future directions for training and local capacity development to support Modulo Basico.

Specific activities included:

- In-briefing meetings with I-TECH Mozambique and CDC Mozambique;

- Introductory meeting with Dra Celia Gonzalez (Adjunct Director, Department of Planning and Cooperation, MOH) and Dra Ercilia Almeida (Director, Department of Health Information, MOH);

- In-depth review of Modulo Basico with the developer Rik Dhaen and other staff at the Department of Health Information who are involved in Modulo Basico (Francisco Macuacua, Alessandro Campione);

- Site visits to a province hospital (XaiXai province) and a health center in Manhica district;

- Meeting with Daniel Lee, a technical advisor in Monitoring and Evaluation seconded to the Ministry of Health's ARV Program, MOH

- De-briefing meetings with MOH and CDC.

We obtained a copy of the application to perform an independent assessment within our software development group as a follow-up to the visit. Also following the visit, we spoke with UCSF about their activities in Mozambique to evaluate facility-level electronic medical record systems HIV care and treatment in use at various Ministry of Health facilities. There is an important linkage between UCSF's electronic medical record (EMR) assessment and I-TECH Modulo Basico assessment, particularly around data quality. Harmonization of recommendations around processes to strengthen data quality within individual patient records at the facility level and within aggregated data feeding into the Modulo Basico system, will be important, and I-TECH and UCSF will remain in communication to assure that we are providing MOH consistent recommendations and technical guidance.

We also spoke again with Mr. Seebregts, specifically about the M-OASIS project's work with the health informatics program at University Eduardo Mondalane (UEM), and about current plans and opportunities of the graduates and students of that program.

Findings

From the meetings described above, and from our analysis of the software and its use in the field, we developed the following findings relevant to the Phase I objectives. We have organized these into description of the scope and use of Modulo Basico, a functional description of the system, and a technical description of the system and its documentation and training materials.

Implementation and use

Modulo Basico is broadly implemented across the country, in all districts and provinces that have electricity. It is currently implemented in about 90 of the 154 districts and in all 11 provinces and at the Ministry of Health Central level. In our own work in other countries, we have found it to be challenging to deploy consistent, uniform systems and practices at multiple sites across a country, and recognize that the current team at MOH has done an impressive job of implementing, maintaining, and managing widespread deployment of the system.

Structure, Programmatic Relationships

The structure of the health care system in Mozambique is composed of four hierarchical levels: Health facility, District, Province, and Central (MOH). Health care centers provide care at the most granular level in areas within a district. Electricity, network connectivity, and medical resources all vary widely across the country. Similarly, there is a variable availability of staff with experience in maintaining information systems. Health centers, health posts, and district hospitals are mostly responsible for data collection and reporting, while strategic planning and management take place mostly at the central and provincial levels. The Ministry is now working towards a more decentralized planning and development model, delegating more responsibility and support to the provincial (and ultimately district) administrative staff.

Many programs contribute to a fully integrated health reporting system (through Modulo Basico), whereby information flows through the hierarchical management structure using a standardized methodology. In addition, there are several vertical health programs, with their own systems and processes for managing health information. One goal of Modulo Basico has been to provide a common platform for handling the reporting requirements across all of the programs, so as to address some of the challenges of supporting multiple separate vertical programs at the health center, district, and provincial levels. The different programs reflected in Modulo Basico each had control to describe the types of information captured in the system; the data elements were not constrained within the application.

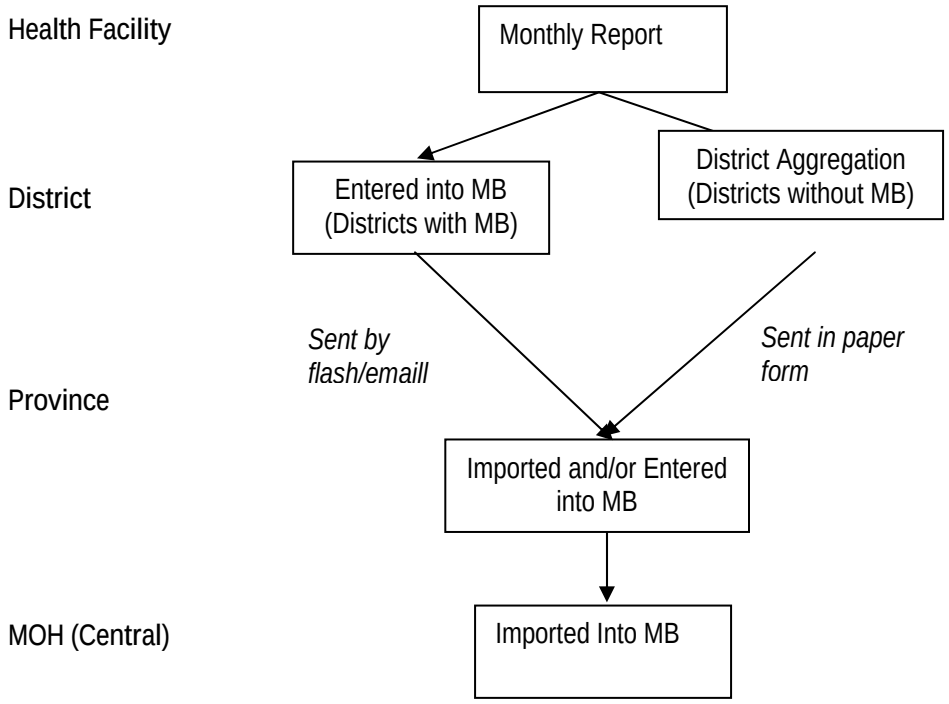
Data entry workflow

Aggregation and reporting of health data in Mozambique begins at the facility level. Summary paper forms are completed by the nurses/medical technicians for various programs and care delivery services (see Appendix 3). The forms are then transferred to the Districts where they are entered into MB (where it exists) by a data entry officer or other staff member. In districts without functional Modulo Basico (due to lack of electricity or other infrastructure), facility level information is summarized on a paper form to generate district summaries.

All health facilities within a district are required to deliver the reports once per month, and in the case of the Weekly Epidemiological Bulletin (BES), once per week. Compliance rates vary widely among programs and provinces. The information on the forms is entered into a local copy of Modulo Basico at the district level, in most cases, though the site code (a unique numeric code for each facility based on geographic and information on facility type) is stored along with the indicators, so that information may be viewed as combined district indicators, or as the individual indicators of each reporting facility within the district. After all health center forms for the month have been entered into the district's copy of MB, the data are exported onto a flash drive, and the electronic data, along with a copy of all the paper forms, are transferred to the Provincial Directorate. If information from some facilities within a district has not been collected by the time of the reporting deadline, districts send incomplete data to the MOH; efforts are later made to substitute more complete data sets when they become available.

The data are then imported into the provincial instance of MB by a designated data entry person or the IT specialist seconded to the Provincial Directorate of Health. In some cases, where for some reason entry of the information into MB at the district level was not possible; the facility forms may be entered at the provincial level. Again, entered data may be viewed as combined indicators at the district or province level, or stratified by individual facility.

Flow of Inforamtion in the National Health Information System



Data transmission

From the provincial level, a backup of the data is made and sent to MOH via the Internet, if network connectivity is available (in most cases network connectivity is somewhat stable at the provincial location). If network connectivity is down, the data is physically transferred via flash drive and taken to the Ministry office in Maputo.

Quality Control

At the provincial level we observed some data quality measures being taken to assess the completeness and accuracy of the data, but we did not find any standardized guidelines or processes.

Modulo Basico contains some reports targeted at monitoring data completeness, as one measure of data quality. These reports are available at the provincial level, but maybe most commonly used at the central level. Phase 2 of this project will include more detailed discussion of the software requirements pertaining to data quality.

Application Description

Modulo Basico has three main functional parts: a data entry interface, data import/export tool, and a reporting interface.

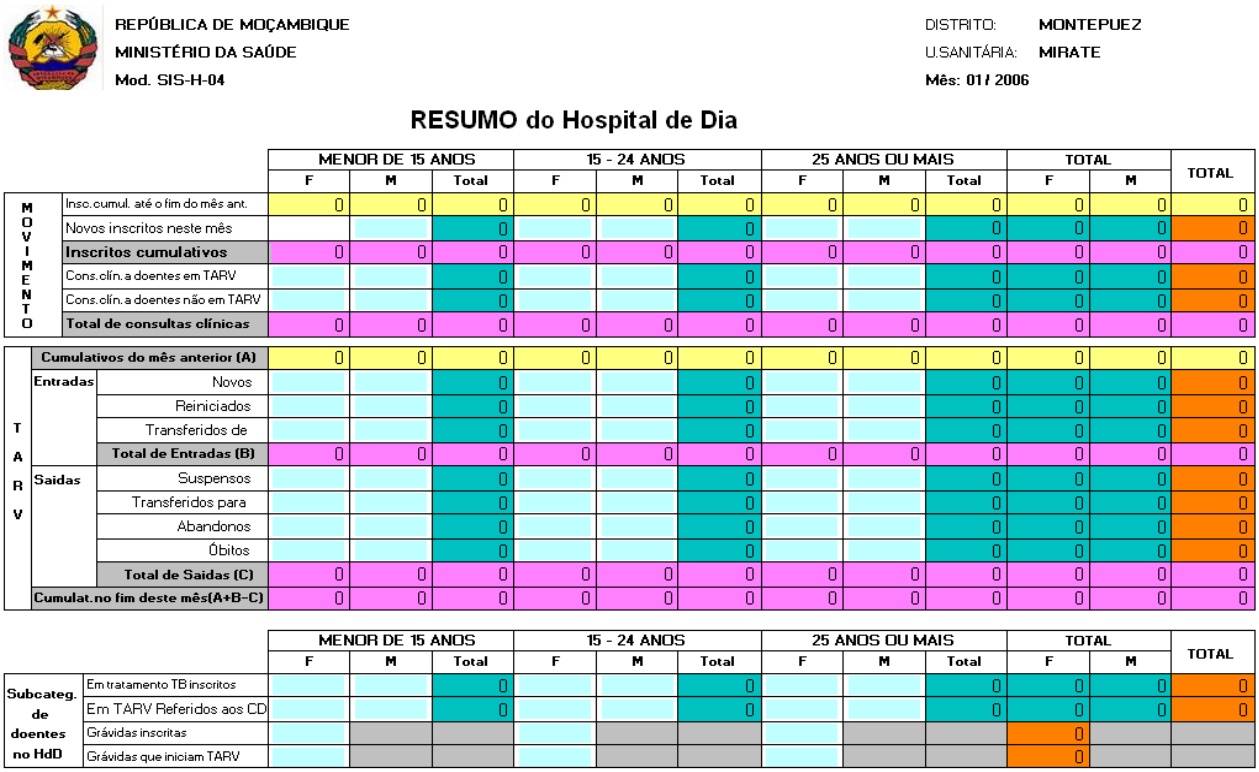
Data entry interface

The success of the application appears due, in part to a clear, simple user interface with a high degree of correspondence between the data entry interface and the paper reporting forms. In addition, data entry is implemented in simple, stand-alone, implementation of the system used at the district and province levels. The home screen consists of a horizontal menu bar at the top of the screen, the title Modulo Basico, and buttons labeled with the coded names of all the forms lined at the bottom of the screen (see figure 1).

Figure 1. Modulo Basico Home Screen

The toolbar includes links to data entry/deletion, import/export, reports, area coding, population information, and administrative system tasks.

The data entry interface consists of screens that mimic the paper forms. The user can tab through the form and enter the data elements as they would enter them on the paper form. The forms all support automatic calculation of totals. Figure 2 illustrates an example data entry screen for a monthly register of health center activity.



REPÚBLICA DE MOÇAMBIQUE
MINISTÉRIO DA SAÚDE
Mod. SIS-H-04

DISTRITO: MONTEPUEZ
U.SANITÁRIA: MIRATE
Mês: 01/ 2006

RESUMO do Hospital de Dia

			MENOR DE 15 ANOS			15 - 24 ANOS			25 ANOS OU MAIS			TOTAL		TOTAL
			F	M	Total	F	M	Total	F	M	Total	F	M	
MOVIMENTO	Insc.cumul. até o fim do mês ant.		0	0	0	0	0	0	0	0	0	0	0	0
	Novos inscritos neste mês				0			0			0	0	0	0
	Inscritos cumulativos		0	0	0	0	0	0	0	0	0	0	0	0
	Cons.clín.a doentes em TARV				0			0			0	0	0	0
	Cons.clín.a doentes não em TARV				0			0			0	0	0	0
	Total de consultas clínicas		0	0	0	0	0	0	0	0	0	0	0	0
TARV	**Cumulativos do mês anterior (A)**		0	0	0	0	0	0	0	0	0	0	0	0
	Entradas	Novos			0			0			0	0	0	0
		Reiniciados			0			0			0	0	0	0
		Transferidos de			0			0			0	0	0	0
		Total de Entradas (B)	0	0	0	0	0	0	0	0	0	0	0	0
	Saidas	Suspensos			0			0			0	0	0	0
		Transferidos para			0			0			0	0	0	0
		Abandonos			0			0			0	0	0	0
		Óbitos			0			0			0	0	0	0
		Total de Saidas (C)	0	0	0	0	0	0	0	0	0	0	0	0
	Cumulat.no fim deste mês(A+B-C)		0	0	0	0	0	0	0	0	0	0	0	0

		MENOR DE 15 ANOS			15 - 24 ANOS			25 ANOS OU MAIS			TOTAL		TOTAL
		F	M	Total	F	M	Total	F	M	Total	F	M	
Subcateg. de doentes no HdD	Em tratamento TB inscritos			0			0			0	0	0	0
	Em TARV Referidos aos CD			0			0			0	0	0	0
	Grávidas inscritas										0		
	Grávidas que iniciam TARV										0		

Figure 2. Data entry screen for a Monthly register form for a Health Center

All fields in highlighted in magenta, blue, and orange will be automatically calculated when the appropriate cyan fields are filled in. Each of the cumulative totals is essentially a stratified (by gender and/or category and/or age) indicator.

Area Coding

Area coding exists to add new health centers, districts, provinces, and even countries to the list of available locations. The user fills in the name of the new location, and an activation code of 1 on the screen and it is saved to the database. The data is then collected for that site without an assigned site code, and the managers at the national level assign it a site code. A particular site must then get an update from the national level with the new site code for the site. Only one addition of a site is allowed per submission of data. This functionality is currently available to all users of the system.

Data Import/Export Tool

A data import/export tool is available to transmit the data to other implementations of MB (e.g., reporting of data from the health center to the district, province, or national level). This tool packages a backup of the access database with all records from the specified timeframe. The backup is saved as a file and can then be transferred and imported to another instance of MB. Figure 3 displays an example export screen.

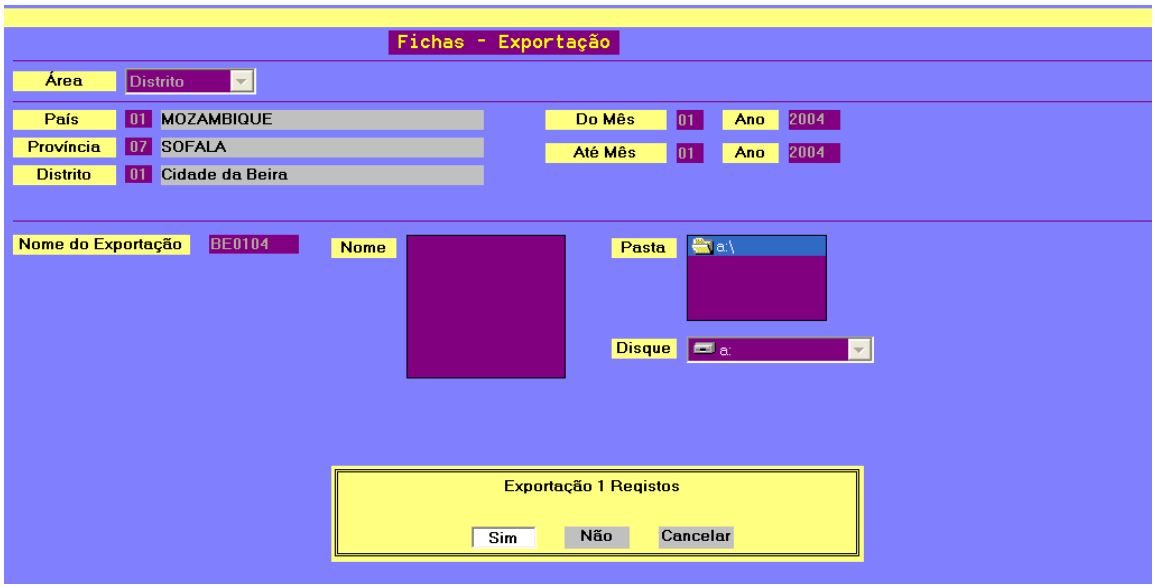


Figure 3. Example export screen.

To export data the following selections are required: Country, province, and/or District and Health Center; the month and year range to of records to export; the name of the file; and the location to place the export.

The import function works by incrementing through each record in the imported file and inserting or overwriting a record into the database that corresponds to the specified form (see Figure 4).

Figure 4. Example Import screen

The import function requires the same selections as the export, and in addition a knowledge of the time range of records in the file to be imported – if a time range is selected that does not correspond to the month and year designations in the specified file only a fraction or none of the data from the file will be imported into the instance of MB.

Reporting Interface

The reporting interface displays excel reports of raw data at all levels that the implementation is authorized to view, and for a specified time frame. There are both standard reports, that display

all the raw data on each form, and a minimal dynamic report generating capacity in which the user can choose the level of data to display (central, provincial, district, health center).

There also exist some application features that facilitate administrative tasks. A user can view/update the population estimates for their region, add/edit/delete health center or district locations, and perform minor administrative adjustments to the system. To date no authentication is needed to access either the database or the system.

System Technical Description

One of the important concerns in addressing the stability of the present Modulo Basico system is the extent to which further development and modification can take place to the software. However, this issue has many complex facets including 1) the underlying portability of the application software and database (it ability to be used on different operating systems and hardware), 2) the stability of the software development environment (needed to modify and rebuild the software application), 3) the ability to modify the installer used to implement the software on a new computer, 4) the underlying technical documentation and its extensibility, 5) the user documentation and training materials and their extensibility, and 6) the skill set and specific experience of the development team with Modulo Basico. These complexities are described further in the sections below.

Architectural and Platform

Modulo Basico currently is written in the Visual Basic programming language, and therefore runs on personal computers that use the Windows operating system. The database implementation requires Microsoft Access 2003 or later, which also depends on Windows.

At present, all personal computers on which Modulo Basico is used also use the Windows operating system. However, it would offer more flexibility if both the software and the database used to develop and support Modulo Basico were "portable", or were able to work on other operating systems in addition to Windows. This would allow for greater flexibility and reliability in deploying the application, and also make possible both a lower cost, open source version of the system, and a self contained or "appliance" version of the system that would be easier to install and maintain.

Development Environment

The computer language used to develop the Modulo Basico system is old, and there has been some discussion among developers about whether it continues to be functional and, if so, for how long. We will discuss the issues around the development software, and the use of that software on Windows Vista, and on Windows XP, the version of Windows which preceded it.

Additional development or modification of the current Modulo Basico software requires use of the Visual Basic 6 (VB6) development environment to modify the code and create a new version of the application. VB6 is an old version of the software that has now been supplanted with Visual Basic.NET. Mainstream Support for Microsoft Visual Basic 6.0 ended on March 31,2005. Extended support ended in March 2008[4].

[4] Product Family Life-Cycle Guidelines for Visual Basic 6.0, http://msdn.microsoft.com/en-us/vbrun/ms788707.aspx (accessed Jan 24, 2009). Also, see http://en.wikipedia.org/wiki/Visual_Basic.

Although VB6 is no longer supported, it remains possible to run the VB6 development software under the Windows Vista[5] operating system It is also possible to run applications developed under VB6 on Vista. Microsoft officially ended support for the development environment on April 8, 2008[6] which means the development environment will no longer function in future versions of Microsoft operating systems, including future versions of Vista. The core VB6 support, which will permit VB6 applications to run on Vista, will be continued for the full lifetime of Windows Vista, which remains Microsoft's current personal computer operating system.

To continue development under VB6 will require using Windows XP, rather than upgrading to Vista. While service pack support (active improvement) has ended for XP, Microsoft will provide mainstream support for Windows XP through 2009-04-14[7], and they will provide extended support through 2014-04-08. According to Microsoft's software support policy, "Mainstream" support includes paid, per-incident support, security updates, non-security related hot fixes, no-charge incident support, warranty claims, design changes, and feature requests[8] and access to information about the product through the Microsoft Knowledge Base and other online support areas. The "Extended" support phase drops non-security hot fixes, no-charge incident support, warranty claims, and feature requests[9].

In summary, while VB6 may be supported under Windows Vista, and Windows XP extended support will continue for another five years, Modulo Basico is currently developed under an environment which is nearing the end of its life cycle.

Installation

The basic setup for Modulo Basico is not fully automated. Following installation of the system files, the application also requires modification of some of the core software libraries supplied with Windows, and it requires the security level for macro execution to be set to "low". These procedures are not fundamentally flawed, but they are somewhat unusual, and require specific knowledge to configure the installation.

The installation consists of a small process that requires a few files (included in the install package). Updates to the installer to incorporate new versions are simple, requiring only the replacement of a single file in the install package. Changes to the database are implemented through the inclusion of a start up script in the application.

If access to the data beyond that supplied by the Modulo Basico application is desired, for instance, to support extraction of the data for graphing or statistical analysis, then Microsoft Access, version 2003 or later, must be installed.

---

[5] Windows Vista is Microsoft's newer operating system, and is standard in computers currently available in the global marketplace.

[6] Support Statement for Visual Basic 6.0 on Windows®Vista™and Windows®Server 2008 ™, http://msdn.microsoft.com/en-us/vbrun/ms788708.aspx (accessed January 24, 2009)

[7] Microsoft Support Lifecycle, http://support.microsoft.com/lifecycle/Default.aspx?LN=en-gb&C2=1173 (accessed January 24, 2009)

[8] While the official communication from Microsoft indicates that feature requests will still be honored for XP until April 2009, we believe that with Microsoft working towards a new release to replace Vista, the XP requests will not receive a high priority.

[9] http://arstechnica.com/news.ars/post/20070124-8691.html

Database Architecture`

MB has a straightforward database design, with a single table per form, and specific fields numbered (or designated by letter) in the order in which they would be filled out. The forms table names are prefaced with 'DATA' and then the short alphanumeric code for the specific form. There are several ancillary tables, including population tables (populated by national census data from the National Statistics Institute) and code tables. The record keys are based on two digit country, province, district and health center codes as well as the month and year of the record entry. Each record corresponds to one paper form for a specific health center, and only one record is allowed per form per month (or once weekly in the case of the BES). Any updates or changes to a form are completed by locating and replacing the record in the database.

Documentation

The system currently has sparse documentation. There exists a training module, a short set of installation instructions, and a technical document detailing the structure of some of the methods in the code (see Appendix 4).

Discussion

In taking on this project, we were asked to a) familiarize ourselves with Modulo Basico's purpose, implementation, and use, b) document existing characteristics of the system, c) identify issues impacting stability and performance of the system, d) characterize the requirements of a system to perform these tasks, and e) develop recommendations for a strategy to move forward with an appropriate course of action.

In the preceding chapters, reflecting the Phase 1 activities, we have described the system and its present usage, functions, and structure. In this chapter, we will discuss a variety of issues impacting stability and performance, along with general principals and recommendations to address them. Then, we will present three basic strategies that may be used, and discuss the priority recommendations assuming that our preferred strategy (incremental development from the current system), is implemented. The Phase 2 project activities are intended to determine and document a series of both system and process requirements, based on further discussion with MOH about their priorities and the tradeoffs and impacts of those priorities on the recommendations in this chapter, as well as the development of an RFP and cost analysis of implementation strategies.

We have organized the issues presented below into several key groups, depending on whether they are mainly software, organizational/implementation, or both. Together, they identify factors which we feel should be addressed to improve specific aspects of the Modulo Basico software, and its implementation and usage, in order to that it becomes more accurate, efficient, useful, and sustainable.

Factors intrinsic to the software itself include:

(1) the architecture of the database
(2) the internal design of the system
(3) the choice of an appropriate application framework
(4) issues of interoperability both internally and with other systems

Factors related to both software and implementation/training:

(5) organizational and technical strategies to facilitate good data quality
(6) effective methods to visualize and analyze the information
(7) a user interface that is intuitive and easy to use by people with a wide range of computer literacy

Factors primarily associated with training, implementation, and usage:

(8) appropriate leadership and governance,
(9) suitable hardware and communication infrastructure and its maintenance
(10) good software deployment strategies
(11) appropriate documentation and training to assure efficient use of, and value derived from, the system by stakeholders at multiple levels.
(12) ongoing end user support to ensure both good data quality, and good use of the data

(1) Database Architecture

The database structure is functional; however, there are several issues which may be addressed to improve reliability, flexibility, and/or efficient retrieval of data. In addition, a restructuring of the data model would facilitate the addition of new data types, or expansion of existing ones, as well as merging of unique data types across forms.

Presently the system uses a combination of fields as the primary key (the identifier which is unique for each record in the database). This primary key should be made completely internal, which allows greater flexibility to store seemingly duplicate information, allowing the retention of deleted or changed records. This is important because (1) it supports auditing, i.e., identifying the person who made specific changes, (2) one can track the rate of corrections made to specific program data or data elements, (3) it permits "rolling back" or resetting the database in the event that significant errors are made in data management accuracy of data can be evaluated, and (4) it allows evaluation of the accuracy of data. When data are updated, it is possible that: 1) the original version represented accurate data; 2) the substitute version represented accurate data; or 3) or both versions contain errors. Retaining data and evaluating significant systematic errors can identify process changes to improve data quality and completeness.

Recommendations

1. *Develop a globally unique internal identifier as a record key.* This supports increased tolerance of inconsistencies when data are moved between different instances, or installations, of Modulo Basico. This allows for simpler data transmission procedures, and decreases the chance of errors which will delete or duplicate data inadvertently. (Short Term)

2. *Modify database architecture to support audit history.* As noted, an internal primary key facilitates support of retained data with status flags to indicate deletion, which is one approach to building an audit history. (Short term)

3. *Modify the database to support imports of overlapping or duplicate data* without requiring explicit and error-prone data management steps (for example, first deleting all data within a date range). (Short term)

4. *Change the database design so that potential linkages between data reported using different forms can be identified.* This is a critical consideration in supporting more complex analysis of data within Modulo Basico. (Long Term)

5. *Abstract data elements from form elements to support reuse of data elements on multiple forms*. This is an important consideration for the extensibility of Modulo Basico, so that forms (interfaces) can be modified in the future when reporting requirements change and so that new types of forms (interfaces) may be added. (Long Term)

6. *Modify the Modulo Basico code base so it can use an open source, portable database*; one that will work both on Windows and on other operating systems, and one for which there are no licensing fees. (Short and Long term)

(2) Application Design

We have not extensively analyzed the internal design of the application, which is to some extent a matter of programming style. However, well designed applications have a clear internal structure, with clearly delineated program functions and clear communication protocols or interfaces between them. In addition, well designed applications are internally documented, both at the interface between components, and within specific functional blocks.

Different languages may lead to some fundamentally different approaches to designing the same application, even when using best practices like those outlined above. In particular, many newer program languages either support or require an object oriented approach to design. We discussed this with the Modulo Basico software implementer at some length. Switching to a newer version of Visual Basic might permit object oriented design, while a shift to other languages, such as Java, may require it. In either case, there is an opportunity to restructure the existing application in a way that will facilitate future development of the application among multiple developers.

Recommendations

There are some specific areas where a high level application redesign may significantly enhance functionality

1. *Redesign the data transfer tool to improve fault tolerance (e.g. ability to transfer overlapping or duplicative data) and support automated notification(e.g., automated alerts via email, SMS, or other messaging which summarize inconsistencies or transfer status).*The code that handles data transfer is a logically distinct function and may be improved independent of the rest of Modulo Basico. (Short term)

2. *Develop additional reporting and data visualization tools.* The reporting interface is presently very simple, and supports export of raw data at all levels, which we believe to be an appropriate strategy. However, the system could be made be more useful with additional reporting and data visualization tools. (Short and Long term)

3. *Reprogram the system with well structured and documented code.* Reprogramming the system will result in improved design, maintenance, and extensibility. This is especially true if the original author of the software is no longer available to work on the system. Writing clean, well documented, well structured code to accomplish the same function illustrated by an existing program is often easier than deconstructing another programmer's work. (Long Term)

4. *Establish a process for authentication and role based access.* The application presently lacks both access control and auditing features. Strict identity management, with authentication by individual user IDs and passwords, timed automatic logouts, and similar features to ensure only the identified user is using the system, are a standard in most health information systems. However, in settings where few people have access to a computer, or where minimal damage could be done from unintended access to the data, these measures may create a significant barrier to the intended use of the system and may not be appropriate. Yet even in the absence of strict identity management, it can still be useful to have users identify themselves. One reason to do so is that role-based access (which limits the parts of the application a person can use based on the permissions associated with the category of user to which they are assigned) can help simplify the interface and guide the user to choices appropriate for their role.Another reason to have users identify themselves, even where access to a system can be controlled through effective limits to physical access (and therefore authentication is not needed for adequate security), is in order to create an accurate audit history. (Long Term)

5. *Develop a secure workflow to transfer data.* Transmission of data using e-mail provides relatively little security, and relatively poor capabilities for authentication and non-repudiation (which ensures that a sender cannot falsely deny sending data). While there are strategies to provide all three of these features through e-mail transmission, they require additional software to use encryption and digital signatures, and add significant complexity to a process better addressed by other methods of communication between applications. (Long Term)

(3) System Architecture/Platform

The present MB requires an old version of Visual Basic (VB 6), which is not natively supported on the Microsoft Vista operating system (OS), which is becoming more common as the operating system on new computers. While this is clearly an issue that needs to be addressed before any substantial enhancement or maintenance work is done on the system, there are two mitigating facts: (a) the compiled version of the system *will* run on Vista – only development is dependent on older versions of the operating system, such as Windows XP, and (b) the existing environment (VB6/Windows XP) can be safely supported for the near term, even though XP will be replaced over time and Microsoft discourages its use. These two facts mean that converting to the current version of Visual Basic does not need to be undertaken emergently, but should be strongly considered to ensure system stability. However, in the long term, the system should be converted to a different development platform, and with that comes the opportunity to move to different system architecture.

Software Platform

The scenarios discussed in this report include both continued Windows proprietary development, and migration to open source platforms. The choice of how to proceed may depend on local capacity being developed within Mozambique, either within the Ministry, or a private business, or through non-profit/academic partners such as the informatics program at M-OASIS. We believe that it's valuable to work with non-profit/academic partners because of both their interest in developing local capacity and their links to similar activities in other countries. The specific choice of architecture depends on local experience and the desire to develop local capacity in one area versus another. However, our experience has yielded good success in developing portable applications using open-source platforms that run under both Windows and Linux operating systems.

Recommendations

1. *Redevelop Modulo Basico using a more flexible and better supported language*. We recommend moving from the current Visual Basic 6 development environment to an environment better supported under Vista, such as VB.net, or to an open-source cross-platform environment such as Java/MySQL, or PHP/MySQL, which is free or low cost, supported under all modern versions of Windows and Linux, and does not require the installation of proprietary software on host machines. (Long Term)

System Architecture

We feel strongly that in most cases applications should be developed as web-based applications, with the client using a browser and the bulk of the application logic contained in an application controlled by a Web server. We feel this is a good architecture even for single user system is running on a local desktop or laptop. In this setting, the application can run on local server software within the computer, and the software may appear to be a local program to the user, but significant advantages remain in using the structure of a web-based application:

1. *Modify Modulo Basico to support a web application framework.* Application development often begins with the intent of creating a local, single-user application, but as the data become more useful, or the workflow around interacting with the data changes, a need develops for two or more users to access the same database. This type of scaling is much more easily accomplished with a web application model then with the traditional single-user application. The performance of Web server applications still lags behind that of a similar problem solved with a single-user, locally run application. However, the performance penalty of the Web has both shrunk, and has grown less important, as Web technologies have improved and hardware performance has increased dramatically. The difference to the user is now minimal, especially for applications like Modulo Basico which depend more on data handling than on computational speed.(Long Term)

2. *Add dynamic web elements to create a rich user experience (e.g., dynamically populated drop downs, client side validations).* User interface development for web applications used to rely on simpler tools than did local application development, severely limiting the user interaction to "browsing web sites". However, the advent of browser-based technologies such as JavaScript, and sophisticated interactions with the server such as those supported under JavaScript enhancements such as Ajax[10], has allowed web applications to offer a rich user experience with dynamic navigation, rapid validation, and other interactive features. (Long Term)

3. *Develop mechanisms to simplify web application deployment*. The complexity of installation for Web application environments (server software, database software, language support, etc.) used to be substantially higher, however as they are increasingly well packaged, this is less and less of a factor. Portable Web applications, which are able to run both in the Windows and Linux environments, are much better suited for distribution as an "appliance", or prepackaged, self-contained application which requires a minimum of technical knowledge to install or administer. (Long Term)

4. *Separate the web application from the authentication module.* Because authentication and security in a Web application may be separated from the application code and handled directly by the Web server, there is a decreased risk of poorly designed code creating security breaches. (Long term)

[10] AJAX is a tool used to create rich Internet applications, it allows web applications to asynchronously request data from the server for display for instance in a dynamic drop down list or grid.

(4) Interoperability

There may be significant value to supporting integration of the Modulo Basico system with other facility level and national level systems. There are current initiatives in several countries to move to a common eHeath national enterprise architecture. These initiatives, two of which are sponsored by the WHO Health Metrics Network and the Rockefeller Foundation, are working to develop standards-based strategies to ensure that data can move appropriately between facility level systems such as electronic medical records, laboratory information systems, pharmacy dispensing systems, and population level systems such as Modulo Basico, DHIS2, CRIS, and others.

Adopting interoperability as a goal means developing shared agreement on both standards of format, or syntactic standards, and standards of meaning, or semantic standards. It also means shared agreement on the procedures and protocols by which data will be sent or made available from one system to another. In many cases the initial, or "reference" applications in this architecture are open source, community supported software, so the efforts to develop interoperability within an application may be freely built on the work of other groups.

Recommendations

1. *Incorporate emerging standards for aggregate data transmission [e.g., World Health Organization (WHO) Indicator Exchange Format (IXF)] into the system design as a method for exchanging data between Modulo Basico systems.* This will not only improve the reliability and flexibility of data transmission at this level, but it will also facilitate electronic receipt of data from facility level systems such as OpenMRS, or iSante, which are actively developing IXF capabilities. It may be appropriate to use IXF as an output from Modulo Basico, if the data from these programs is sent to CRIS, or other systems used at a national or international level, and which support IXF. (Short and long term)

2. *Investigate and align with any other standards used nationally in Mozambique, or emerging as part of international initiatives*. For areas such as identity management, authentication, data security and transmission, measures and indicators, or other areas where there is active work which can be leveraged. (Long term)

3. *Investigate standards which may support compatibility with future, integrated informatics applications (e.g. financial, human resources, supply chain management, donor reporting, or others).*(Long term)

(5) Data Quality

There are both organizational and technical strategies that may be used to improve data quality. Some elements of quality assurance can be implemented centrally, but quality assurance is most effective when the data are used close to their source, and the quality of the data are audited and maintained primarily at that point. Therefore, data quality is closely tied to the goal of making the data useful as close as possible to the point of entry. To do this effectively may require new training for existing staff, new features to support the use of the data at the district or provincial level, and possibly new roles at these levels to make better use of the information in the system.

Recommendations

There are presently reports in Modulo Basico which allow comparisons of some aspects of data quality between sites.

1. *Examine the present data quality metrics, and develop new metrics as needed to describe the completeness, timeliness, and accuracy of data entered between districts and/or provinces.* These metrics should be well described, clearly documented, and available in system reports for easy, ad hoc, characterization of quality. (Long term)

2. *Incorporate procedures for using the aforementioned metrics, including frequency of review and strategies to improve performance, into the training for appropriate district, provincial, and central staff.*(Long term)

3. *Better characterize the timeliness and gaps in reporting.* As part of an ongoing data quality improvement process. comparison data from other organizational units should be made available for benchmark comparisons by district and provincial staff.

(6) Information Visualization and Analysis

Effective methods to visualize and analyze data collected using Modulo Basico may help substantially increase the value of the data to those most directly involved in collecting it, and hence help increase the quality of those data, however these enhancements may also substantially add to the value of the data both centrally at MOH, and within the individual programs at MOH.

Modulo Basico contains the ability to export data in a simple tabular format, or to retrieve the data through Microsoft Access. However, other techniques can support access to the information in Modulo Basico with less technical training.

Recommendations

1. *Integrate a reporting package with Modulo Basico.* The package should capable of tabular and graphic display of data, and of exporting data into common formats usable by a program such as Excel, Access, and SAS. (Long term)

2. *Work with users of Modulo Basico at the district, province, and central level to develop a small member of useful visualizations and analysis data set formats as examples*; which would then be implemented in the reporting package, for example, period provincial reports, including basic tables and graphs. Users could build on these examples and create further visualizations or analysis data sets useful for their specific program or in their specific environment. Short term)

3. *Develop a specific strategy to share these sorts of user developed reports, visualizations, and data sets along with an e-mail list, forum, wiki, or other tool to support comment and collaboration.*(Short and long term)

4. *New training for the data managers at the provincial, and eventually district level, to support effective use of these data.*(Long term)

5. *Produce reports in a format compatible with the web portal software used by MOH to facilitate dissemination of reports*(Long term)

(7) Usability Design and Testing

Modulo Basico is a straightforward system whose ease of use is largely attributable to the extent to which the user interface mirrors the paper data collection forms. Another positive and important aspect is the rigidity of the system, in the sense that it does not allow much intervention (manipulation?) on the part of the user, which is a success factor in the Mozambican context, as demonstrated by other experiences. This is an important concept to retain, as the system may often be used by people with a minimal amount of training, or those who are trained by their peers.

At the same time, it is valuable to take a more formal look at the functions of the system, and at the way in which users address those functions. Often this can give very valuable insight into the structure of navigation through a program, the importance and availability of particular features for obtaining online help with the program, flaws in the data entry process that may lead to inaccurately entered information, or other problems. As system capabilities increase, for instance through adding visualization and analysis data set capabilities, or tools to support data quality, it is important to develop test scenarios that describe the use of these features.

Recommendations

1. *Redesign Modulo Basico user interface to address its graphic elements, navigation, and layout of the non-form related user interface components*. (Long term)

2. *Preserve the parallel between the forms and a user interface for entering data off of a particular form, and keeping or increasing the validations associated with entering specific data elements incorrectly*.(Long term)

3. *Develop a workflow that allows a data entry person to mark a form as needing specific review or reevaluation due to data which appear to be correctly transcribed, but which do not validate correctly.*(Long term)

4. *Develop scenarios that describe use of the common and major functions of Modulo Basico.* These scenarios should be used in observed usability trials to identify specific barriers to correct use of the system that may be addressed by redesign. (Long term)

5. *Develop scenarios to support standardized, formalized testing of new versions of the software to ensure that errors are not introduced by the development of new features.*(Long term)

(8) Leadership and Governance

The experience acquired, and relationships developed, in the process of training, deployment, and operations of Modulo Basico are significant leadership assets for the program. We believe it is important to build on the existing relationships, staff, and procedures for the continued deployment, development, and use of Modulo Basico.

Governance

While MOH has placed the leadership of the program within the DPC, large programs like Modulo Basico which involve multiple levels of leadership and provide information and useful across a wide range of programs, can often benefit from an advisory council or steering group to provide guidance for the overall program.

1. We did not assess the presence or interest in such a stakeholder group on our first visit, but would be interested in exploring whether this would be appropriate within the organizational structure of the system.

Technical Leadership

The team within DIS involved with MB has been closely associated with the development and guidance of the program, both at the MOH level, and in providing training and support around the country. Their viewpoints regarding the characteristics of MB that permit its widespread dissemination (simplicity of installation, operation, relatively insecure access controls, etc.) are very valuable.

1. *We believe it would be valuable both to retain the practical experience of the persons in the original team that developed MB, while also developing a sustainable structure for technical leadership within the Ministry.*(Short term)

2. *Ultimately, local experts should lead the project*. For now, continued external technical assistance should be maintained while the capacity of local systems and staff are strengthened in the technical management and leadership of MB. (Short and Long Term)

Resources (Central Level)

For the system to be stable, there must be an informatics engineer (or technical leader) who is familiar with the system, and the capacity to change and develop the system should be housed within, or directed by, MOH. This requires further identification of, or development of, technical skills within Mozambique. The constraints of specific staff availability and expertise among the original MB team will dictate how best to identify and develop this capacity, and how to support the system in the interim. Staffing will be addressed in more detail in phase 2.

Resources (District and Province Level)

We did not do a comprehensive inventory of the human resources dedicated to MB at the district or province level. However, it is important that adequate support for the use and maintenance of the system be provided at these levels for the system to be properly utilized. It may be of benefit to explore additional utility of the data at these levels as sites have the greatest incentive to improve data quality when it has benefit to their own operations. Appropriately, while we understand that a significant portion of the present training focuses on acquisition of basic computer skills, our recommendations highlight the importance of developing data management and use skills at the district and province level.

Recommendations

1. *Staffing should be sufficient at the provincial level to provide IT support to all district sites within the province as well as fulfill reporting requirements to the Ministry.* Staffing should also be in place to provide training and user support for the application. (Short and Long term)

2. *Appropriate resources must be in place to support existing staff and train replacement staff at the central, provincial, and district levels.* (Short and Long term)

3. *There must be new deployments in the districts which currently do not use the system, including the training of new staff, carrying out installations and upgrades, and providing hardware maintenance and other technical support*.(Short and Long term)

(9) Hardware and Communication Infrastructure

Modulo Basico is a fairly "light weight" application, or one that does not make extensive performance demands on standard PC hardware. It does require the Windows operating system, but does not require an Internet connection. Our recommendations focus on preserving this hardware independence, while allowing for improved packaging of the application, and on positioning the application not to depend on the Internet, but to be able to take advantage of connectivity as it arrives in new locations over the next several years.

Computer Hardware

The systems on which Modulo Basico is installed are standard desktop PCs running the Windows XP operating system. These systems can be challenging to maintain as they are prone to hardware failures, may have viruses or other installed software corrupting the operating system and/or causing security compromises, and subject to a wide variety of user installed software which can cause many of the same problems.

Recommendations

1. *We recommend that Modulo Basico, at least at the district level, continue to be a lightweight application with minimal constraints on the operating system software or hardware.* The system should not depend on specialized hardware or expensive "server level" operating systems, particularly at the district or province levels. Depending on the application development platform chosen, a self-contained or "appliance" version of Modulo Basico may be developed which can be more easily shielded from the risks and inconsistencies of the system on which it is running.(Short and Long term)

2. *The Central level should have better redundancy of hardware*, with a second system available in case the primary system experiences problems, and it may need somewhat higher performance hardware than the provincial systems. This will depend a great deal on the application development choices which are made, and cannot be economically determined in advance. (Short term)

3. *A protocol should be developed and adhered to that will sufficiently protect client machines from viral infection.* (Short and Long term)

Data Transmission/Communications

Modulo Basico relies on the transmission of electronic data via either e-mail or file transfer using media such as USB sticks, and on the transmission of paper as a backup for the electronic media. As discussed previously, e-mail attachments are insecure, and it is a challenge to implement validation and authentication/non-repudiation through e-mail. That said, e-mail is relatively reliable, and tolerates poor, low bandwidth connections relatively well.

While many of the district facilities are not connected by reliable Internet, it is better to design for the near future and accommodate the present, rather than focus too strongly on unreliable connections.

Recommendations

1. *Develop a file transfer of signed, encrypted data be that is supported in a media independent way, to include USB sticks, CDs, or manual e-mail attachments.*(Short term)

2. *Develop automated, scheduled electronic data transmission,* using methods such as sftp (secure file transfer protocol), or https (the encrypted protocol used for Web transmissions) where reliable or intermittent Internet connections exist,. (Short and Long term)

3. *Paper transmission of the forms should be continued* both as a backup procedure and as a way to validate data at the district and provincial levels, where appropriate. (Long term)

(10) Software Deployment

Modulo Basico is presently developed with an ad hoc release schedule, which appeared driven primarily by the need to incorporate new forms and data into the system. We did not assess, in detail, the way in which new releases get distributed, nor did we assess how uniform was the use of the latest distribution throughout all districts.

1. *An ad hoc release schedule may be appropriate*, though as the development path becomes more complex and the software serves more users and more programs, supplying more features, it may be desirable to switch to a different development methodology based on periodic rapid releases. (Short and Long term)

2. *The data updates generated by the software should be internally labeled to indicate which release of the software generated them*. As the size of the deployment becomes more complex, managing conversions between different "generations" of forms should be as automated as possible. We have had good success in using this type of internal labeling to track deployment of an electronic medical record throughout multiple sites in another country. In this case, we have found it very helpful to monitor how quickly new releases were deployed at the more difficult to reach sites.(Long term)

3. *To the extent possible, distribution or new releases should be through an automated update mechanism.* .Methods for distributing new releases vary with the application development strategy used, with the data transmission mechanisms available,

and with whether or not automated update mechanisms or an appliance model can be used.

(11) Training and Documentation

It is important to develop, document, and teach protocols and procedures for the use of Modulo Basico at all levels, including both data entry and data use, as well as for administrative functions such as system operation and maintenance, security, data transmission, and quality control. Some of the principles that apply to training materials also apply to documentation: that it is task oriented, describing system functions for specific user roles. Both training materials and documentation should be available in both printed form and electronically, both readily available where Modulo Basico is used.

Training

The development of training materials and the delivery of training are complex topics somewhat outside of the scope of our original Phase 1 assessment, but both are critically important for the ongoing success of the system.

Recommendations

1. *Training materials should be developed based on a set of common usage scenarios*, which typically are the same usage scenarios used for usability testing and user interface development, as well as for new release testing. The training materials should be functionally oriented around the tasks that are completed within the system. (Long term)

2. *Training materials should be made available electronically, and the content of training courses should be captured and distributed on CD or DVD*. It is desirable to have an asynchronous, off-line method of distributing training materials, particularly in a large country like Mozambique with significant travel times and remote areas. (Long term)

Documentation

There is relatively little documentation for the existing system (see Appendix 5). There is a user's guide/tutorial, which will be reviewed and updated.

Recommendations

1. *A system overview document should detail processes and types of information flows handled by current Modulo Basico (MB), including who and how people interact with the current software, and the relationship between paper-based and electronic information flows.* An architecture overview/data base design document should be developed which provides an overview of the data model and database, as well as an overview of the design and function of the software. This document should also provide a mapping of forms elements to the data model and documentation of the support tables, as well as of installation, troubleshooting, and support procedures. (Short term)

2. *The detailed software documentation comment describing the individual software components and exchange of data between those components should be contained within the source code itself.*(short term)

3. *User documentation should be functionally oriented and should be linked into the software through a help system*. The content of the user documentation should be distributed with releases of the system, so that the documentation is always available. Documentation of system use should be comprehensive, including both processes to obtain the summary data at the district or provincial levels, as well as the procedures to enter those data into Modulo Basico. (Long term)

4. *Documentation for system processes such as installation, maintenance, security, data transmission, and data quality monitoring should be developed.* (Long term)

5. *Indicators and measures should be documented*. This can be useful within the country as well as internationally, comparing indicators used for clinical decision making as well as programmatic needs and strategic planning. While this can be done as part of having agreed-upon definitions and methods for ascertaining data collected by Modulo Basico, the documentation may have value in shaping program assessments as well. These indicator definitions could be managed as part of an indicator repository, with centralized procedures for review, documentation, and collection or calculation, managed within MOH. (Long term)

(12) End User Support

While training and documentation are vital to supporting end users in entering high quality data, as well as in making use of those data, it is also important to provide direct support to those users. There must be a way both to address questions that arise about the use of the system or about problems with the data, and there must be a way to obtain prompt technical assistance in the event of software or hardware failures.

Recommendations

1. *We recommend that, depending on the available technology in a district, direct "help desk" support be available through both email and telephone*. This could be structured either through the Provincial Directorate of Health, or through the Ministry offices in Maputo, depending on communications and Ministry staffing considerations. However, we recommend that this function be available centrally, even if some districts may choose to consult the Provincial Directorate of Health first. (Long term)

2. *We recommend that a single point of contact, as described above, be used both to answer questions about the use of the system, and to initiate onsite technical support in the event of software or hardware failures.*(Short and Long term)

3. *We recommend that direct support to the district sites continue to be provided through the Provincial Directorate of Health*.(Short and Long term)

Primary Options and Priority Recommendations

Broadly speaking, there are a range of scenarios for the future of the Modulo Basico system in Mozambique. Choosing between them depends to a large extent on the availability of different

human resources, financial resources, and capabilities, and to some extent on the planning horizon and on the vision for a national approach to integrated health information systems.

Primary and Secondary Goals

We believe the primary goals of Modulo Basico development should be to ensure a system is in place which meets the core requirements of data collection, standardization of information, quality management, and data usage, and which is aligned with the future, and perhaps evolving, vision of MOH.

The most important secondary goals, in our opinion, include

- flexibility in the development plan,
- local capacity development and incremental assumption of full responsibility for the system,
- clear documentation and best practices in development of the system, and
- good training to ensure that the system is used efficiently, that the data are accurate, complete, and timely, and that the users are able to ensure that the information in the system can find its way out – that the system is useful and valued.

The feasibility of an approach to any specific issue in addressing those goals will be determined by a variety of factors including specific constraints of human or financial resources, the national priority placed on goals such as local capacity development, alignment with international initiatives in information system design and implementation. The eventual priorities placed on these addressing these sometimes conflicting priorities may substantially change the optimal recommendations. However, notwithstanding those constraints, we will further discuss recommendations we feel most appropriate for the incremental approach.

Three basic scenarios

Three distinct alternatives to move forward in the development or replacement of Modulo Basico are:

1) Maintain the current Modulo Basico software application while enhancing system documentation and providing further training of system managers and data managers at the central and provincial levels to ensure system stability and maintainability. This is the least expensive of the basic alternatives, particularly if there is continued involvement of the original system developer. Limitations of this approach include both the stability of the current development platform and the challenges in extending the application to improve functionality in key areas such as dependency on one primary programmer, user authentication, features to support data quality, and visualization of data for analysis. The platform stability could be addressed by migrating the software to a more current Microsoft software development environment, and features could be added to the current system, but with the result that the cost of this option would increase sharply.

2) A second possibility is to abandon the current Modulo Basico application and replace it with an existing open-source or commercial software product in the same niche. This

option has significant benefit for improving the functionality of the application, or for choosing a replacement application intended to fit into a national architecture of facility level, district level, province level, and national level interoperating information systems. However, this option has a very significant risk of failure. Not only is there the usual risk of issuing a request for proposals (RFP), negotiating requirements with vendors, and executing a commercial implementation, but there is a significant risk of disruption in data flow, loss of data, and burden to customize or configure an existing product to meet the specific needs in Mozambique which Modulo Basico has addressed. Implementation of an open source product, if there is a local committed group to support that implementation, mitigates some of these risks. There is at least one well-developed open-source product available, DHIS2 (see http://www.hisp.org/), which is being used in several other countries as an equivalent system to Modulo Basico and part of a larger national architecture. However, there was a previously unsuccessful implementation in Mozambique with a much earlier version of this software, which may make for greater challenges in working with the core development team. Moreover, any rapid migration to any new product, whether commercial or open source, requires very significant resources for training and technical assistance, and some software may require significant investments in hardware and communications infrastructure.

3) A third possible approach is to take an incremental approach to the extension and redevelopment of the Modulo Basico application. For instance, expanded functionality such as visualizations, or improved exporting to statistical or other software packages, may be most important at the central level. The same may be true of issues impacting reliability, performance, and access controls, since the central level accumulates the greatest amount of data. At the same time, the central level is the easiest area in which to develop new software both because the infrastructure is better developed than it is in the smaller cities, and because there is only a single production system at the central level. An incremental approach would involve minor changes to Modulo Basico to improve the development stability of the present version, and add features for more automated and reliable data transfer, perhaps using international standards such as the WHO's Indicator Exchange Format (IXF). That would permit the replacement of the central system with an open source product such as DHIS2, and the development of local capacity around the support and use of that product, within MOH, within the informatics group at UEM (M-OASIS), a locally contracted external consultant/company, or a combination of these. Enhancements such as access controls, auditing, data quality management, and visualization are either already part of the DHIS2 system, being worked on by open source contributors to that project, or could be developed for Mozambique and contributed back to the DHIS2 community. The health-care facilities would continue to use paper, and the district and provincial levels would continue to use the current standalone Modulo Basico application, with the data transmission changes described previously. As the central system matures, copies of that system could be moved to the provincial level, while leaving simpler data entry intact at the district level. Similarly, the next increment, once there is a good deal of experience with the new system and the risk is significantly reduced, would be to move that system out to the district level. Several years from now, with improved communications in some districts, and the anticipated increase in use of facility level electronic medical records and electronic patient tracking systems, moving a system such as DHIS2 to the district level will greatly facilitate automated data exchange with facility level systems -- part of the vision of the national eHealth enterprise architecture initiatives currently supported by Rockfeller and the Health Metrics Network in Sierra Leone and Rwanda, and by the IDRC in Mozambique.

We recommend the incremental approach as a middle ground in terms of risk and cost. It provides a path forward without mandating a particular speed, so the national implementation plan can be adjusted based on available resources, changes in local capacity or external assistance, and experience. It has the significant advantage of letting Mozambique benefit from progress made on other national initiatives in Africa, and beyond. We feel this approach will minimize risks for disruption in data flow, will support the progressive development of the skill base of local system managers within and/or outside of the Ministry of Health, and will allow alignment with forward-looking goals of the Ministry of Health for a more integrated, efficient, and sustainable national health information system.

Incremental approach

Below we describe recommendations, consistent with an incremental approach and specifically with regards to application development, system administration and IT support, user training and documentation, data validation, and reporting and data utilization. In each of these areas, we focus most attention on recommendations regarding the software itself and the application development process. Of course, there are many resources required for successful deployment and use of any software, including appropriate equipment and technology and sufficient human resources with appropriate skill levels. While we make mention of these resources here, or in the recommendations section, our recommendations do not reflect a comprehensive evaluation of the current status of these resources nor of future resources needs. Policy and decision-makers will need to consider our technical recommendations with an understanding that we have not attempted to specifically identify resource requirements at this phase.

Application Development

Several of the priority recommendations cluster around application development, which must be addressed in order to move forward. I-TECH/UW is available to provide close technical assistance and support to any of the parties engaged in continued development, both in the areas of database architecture and application development.

Our priority recommendations are:

1. The issue of staffing the software development team or identifying a specialized company for outsourcing parts or all of software development and maintenance is critical, and needs to be addressed promptly as many other decisions will depend on the skills and interests of the group selected. Possible development staffing options include:

    - Identify a group with experience in informatics in Mozambique (for example, private companies or M-OASIS/UEM Informatics Program which uses recent graduates and students as a long term solution), with external technical assistance for short term development and overall design and architecture support..

    - The team involved in MB at DIS to continue as the anchor of a team redeveloping the software in VB.net, to capitalize on his familiarity with MB system, again, with external technical assistance as needed.

    - MOH to identify and develop several technical staff, either internally or as new hires, to build the capacity for continued development of Modulo Basico and similar projects. Knowledge management in software projects is a challenging problem. While it would be ideal if all aspects of the system were documented so a new

programmer could pick up a project with little effort, in practice it is important to have a multi-person team, typically with a diversity of expertise and career levels, so that most of the Project knowledge is retained as individuals come and go.

Our preferred option, for the short-term, is to use I-TECH/UW to support Modulo Basico in continuing primary development, under the direction of ministry staff involved in the project. In the long run, we believe that it makes sense for MOH to create a strong software development capacity within the organization as the efficient delivery of healthcare and the effective management of health care systems both depends on rich and accurate information made available to clinicians, administrators, and government leaders. An effective way for MOH to build this capacity may be to use M-OASIS or other local graduates and students to support the system initially, and to bring those people into the Ministry as the goals and vision for healthcare informatics solidifies.

2. Migrate Modulo Basico to a new development environment compatible with the newer Windows operating system and capable of cross-platform deployment.

   This would involve technical assistance to establish a new development environment and selection of a team of local, in-country developers. Depending on the development environment selected and the existing skill-level of the local team with various programming languages, significant hands-on training may be required for the local team to efficiently and effectively work in a new development environment. Possible development environments include an open-source environment (using Java programming language) and the Windows environment (using VB.net programming language). The M-OASIS/UEM informatics group, through its collaboration with the IDRC-funded OASIS project and its technical assistance from the University of KwaZulu Natal in South Africa, has developed some expertise in the open-source development environment using Java. There will be many advantages over the long term to Mozambique's health information system to be able to deploy, adapt, and support open-source software tools.

   The choice of application development platform is somewhat dependent on the choices made regarding staffing, but includes:

   - Continued exclusive use of Microsoft platform, but updating the system to VB.Net.
   - All additional work done in an open-source environment such as PHP or Java

   Our preferred option is to plan for some additional work on the present VB6 software, continuing to use the Windows XP development environment for this activity. The additional work would focus on improving the interoperability and data transmission features of the current version of the software. At the same time, we would begin working on a more robust version, based on the Java language, which could incorporate features previously identified as important. That version would be initially deployed at MOH centrally and, as it matured, could be deployed in the provinces and eventually, using an appliance model, replace the systems deployed at the district level. As mentioned, we believe that it is important that Modulo Basico remain a simple, stand-alone application from the perspective of the district.

3. The development process should be transparent and well-described, and which helps develop the skill base and expertise of a team, rather than of a single individual. This

recommendation should be implemented regardless of the composition of the selected in-country development team and the specific development environment selected

4. Redesign the database architecture so that it better supports extension of the system in the future as health reporting needs change and better supports efficient analysis and visualization of data. Key elements of a revised database architecture are: the use of a hidden, internal primary key, the linkage of similar data elements across forms, and the ability to handle deletions and updates by a marketing older records, rather than overwriting them. These elements support the goals of extensibility and data analysis capacity by allowing for indexing of data in ways beyond the current site/date presently used, by allowing reports to be produced that more easily combine data from multiple programs, and by allowing for the maintenance of an audit history, which is a best practice for information systems that are based on transactions such as Modulo Basico.

System Administration and IT Support

This section will discuss a set of features which add specific functionality to the application or its distribution, and which impacts the installation, administration, security, and maintenance of the system.

Our priority recommendations are:

1. We've discussed several ways to improve data transmission. Perhaps the most important of these are the changes related to attaching a unique identifier to each row of the entered data, so that databases may be merged without human attention to deleting potentially duplicate data. This capability depends on the database changes discussed in the previous section.

2. The specific features which can be added to support improved system administration will depend on the choices made for Application Development. However, we feel that adding login information, even if this identity information is not enforced by passwords, is important as a way to begin to track actions taken by specific users. Similarly, even if users can declared their identity with weak or no protections, allowing them to select a role will simplify training by limiting the options available to a particular type of user.

3. Changes to the workflow and format of the data extracts created to transfer data to the next higher level can be modified to become a backup mechanism as well, which decreases the chance of data loss as well as a site's ability to recover from errors.

4. Further assessment of the administrative and support requirements at the district level, and the development of a plan to ensure that adequate staffing and time are available at these locations.

5. Provide some collaborative exchange of information between the provinces to encourage sharing of information about how Modulo Basico is maintained and used

Important recommendations which will take longer to implement include:

1. We are working now on an "appliance model" of software distribution in our projects in Haiti. We believe this method shows significant promise for distributing Web applications as the application software, database program, Web server, and supporting

software can all be bundled together and distributed as a single executable image, which then can be accessed via a web browser for both administration and routine use. The use of these methods in Mozambique, as they are proven elsewhere, will significantly increase the ease of distribution and decrease the installation effort of Modulo Basico.

2. Implementing standards-based data transfer is an important first step towards replacing the system used at the Ministry with a more robust data warehouse environment, whether that be open-source, commercial, or developed specifically for Mozambique. In particular, using IXF as the exchange format may allow integration of Modulo Basico with similar systems are data warehouses being developed in other countries.

3. The needs at the district level are primarily data entry and validation, while the needs at the central level are primarily integration, retrieval, analysis, and visualization to support the leadership and strategic planning. While those two distinct sets of needs are currently served by one system, the difference between them also provides a path for a long-term evolution of Modulo Basico. We believe that with the expanded uses that the central level are best served by a data warehouse system, or substantial development of Modulo Basico including changes to the data model, architecture, and functionality. At the same time, those changes are much easier to implement with a single instance of the system. Therefore, we propose that the development be done on the central system to increase the functionality. If the transfer of data between these different systems is standardized, then the new central system can eventually replace the provincial systems, giving them substantially more capability, as it matures. Looking ahead to better connectivity and computer skills at the district level, this data warehouse approach may find its way to the district level eventually, but all this can be done at that pace dictated by the needs and skills at different levels. In summary, we recommend that new functions be incorporated by diverging the software development path to account for the two very different ways in which the system is used, with the eventual goal of migrating rich functionality, and the knowledge to use it, throughout the country.

User training/documentation

While outside the scope of software development, ensuring that there is adequate training on the use of the system and on the use of the data organized by the system is essential to ensure both high quality data and its effective use.

Our priority recommendations are:

1. Edit the user's guide to ensure that it includes all functionality and screenshots of the current system.

2. Support improved utility of the data by working with MOH to identify and develop a set of visualizations, both to make clear the quality and consistency of the data, and to provide insight as to the reported indicators.

3. As part of a substantial overhaul to the application code, we would recommend adding context-sensitive help, so that users are more likely to see health information that relates to the task they are currently performing.

4. The existing training materials should be expanded to support in person, direct training as well as asynchronous training with video clips and examples distributed through media such as CD-ROM disks and You Tube.

5. Develop and implement a training plan associated with the launch of a re-developed MB, and which supports future training needs as new features and functionality are added through the incremental approach. There are a variety of types of users, and it is important to distinguish the unique training needs of each type so that training resources may be used most efficiently. Those who interact with the system at the health center and district level constitute one group of users, who need to master user login, data entry, data validation, data transmission processes, and basic data analysis. At the province level, system users may need to have additional expertise in system administration issues and more advanced data analysis. At the central level, users need further expertise in data integration and system modification.

6. Enhance availability of site-level technical assistance to support training and orientation of all users in evolving features and processes of information system. Modulo Basico and other health information system components exist within a dynamic global environment for information and communications technology. These systems can and must continue to evolve to take advantage of innovative technologies when these offer advantages of increased utility, lower cost, and easier maintainability. Therefore, it is important to consider training not as a one-time need specific to a particular software product which remains static over time, but rather consider training as a nimble function which efficiently supports changes in processes and systems over time. A cascade model whereby health information system specialists from the central level can train personnel from the province levels, and province level specialists can train users at the district and facility levels, will be advantageous.

7. Take advantage of the experience and personal connections developed by the DIS development team. This could involve a variety of roles from continuing to travel and lead implementation and training, to collaborating on training materials development.

Data Validation

Data quality is critical to both effective use of the system at the central level, and to the recognition of the value of the system at the provincial and district levels.

Our priority recommendations are:

1. Develop a capability so individual data items and forms can be flagged for review if there is uncertainty about the data that are being entered.

2. Integrate the data review capability with data element level validations to screen for different types of errors are inconsistent data.

3. Develop reports, in addition to the current reports, which characterize and visualize gaps in data completeness or timeliness.

4. Develop a set of district and provincial level procedures for data quality review.

In describing the above recommendations, we went to acknowledge that this will require close cooperation with the individual programs

5. identify a programmer level data quality contact to monitor data quality for each program centrally, and to participate in decisions about levels of validation and data quality procedures.

Phase2 Objectives

Phase 2 involves completing detailed requirements gathering for an implementation of an aggregate reporting infrastructure in Mozambique. This will likely involve multiple site visits to various provinces that differ widely in their programmatic needs as well as their availability to medical, network, and power resources. We will focus on documenting in detail the types of data that MB handles, detail the system architecture, as well as the hardware and software performance on site. We will also perform an inquiry on site to discuss reporting and analysis needs for programmatic support at all levels. In addition, we will be documenting in detail the linkage between the paper system and MB.

In addition to system requirements, we will focus on documenting process requirements in the following domains:

- Participatory design, Usability Development/acquisition/configuration
- Testing Deployment Maintenance and support
- And Curriculum development and training

Deliverables for the Phase 2 objectives will include an RFP for use by the MOH to bid out work/recommendations, and detailed cost analysis of implementation options,

Phase 2: Detailed Requirements Documentation (January 2009- March 2009)
Objectives
Complete detailed requirements-gathering through further in-country assessment.
Document system requirements, via use cases or other methodology, in the following domains:

- Type of data handled in MB system, data collection and data entry process
- Data management and quality assurance
- Reporting and data analysis processes and functions
- Role of paper-based information systems
- System architecture, hardware and software performance
- Interaction with electronic information systems used for other purposes in the health domain, including common data exchange formats

Document process requirements, as appropriate in the following domains:

Participatory design
Usability
Development/acquisition/configuration
Testing
Deployment
Maintenance and support, and staffing
Curriculum development and training
Financial analysis of costs associated with each proposed strategy

Appendices

Appendix 1 – Proposed Work Plan/Timeline (as of September 15, 2008)

Modulo Basico Assessment
I-TECH Mozambique Proposed Work Plan

Phase 1: Initial Assessment (October 2008-January 2009)

Objectives

1. Document process and type of information flows handled by current Modulo Basico (MB), including who and how people interact with the current software, and the relationship between paper-based and electronic information flows.
2. Document key aspects of Modulo Basico database architecture.
3. Identify conditions, resources, and inputs required to sustain Modulo Basico stability in coming 1-2 years, including human resource capacity needed for system maintenance.
4. List types and scope of additional system documentation necessary, given scenario of continued use of existing Modulo Basico in coming 1-2 years.
5. Identify performance gaps, including gaps in data quality or timeliness, associated with current Modulo Basico.
6. Describe recommendations for updated, enhanced system functionality to meet MOH's needs.
7. Outline possible system architecture and associated IT platforms for redevelopment of Modulo Basico to achieve these recommendations, with general cost and risk estimates.
8. Outline 2-3 scenarios for scope of redevelopment of Modulo Basico (which may ultimately be undertaken in a phased manner), with general cost, time, and risk estimates.

Activity Description

I-TECH will conduct an initial 2 week visit to Mozambique in late October-early November. This visit will include briefing meetings with the MOH and other stakeholders as well as in-depth meetings with the developer of the existing MB Access database. The visit will also include district-level site visits to observe use of the existing MB system and learn about needs. Following the visit, I-TECH will compile a report, and will facilitate a half-day videoconference to review the findings and recommendations with MOH and stakeholders.

Deliverables

- Report covering Phase 1 objectives

Phase 2: Detailed Requirements Documentation (January 2009- March 2009)

Objectives

1. Complete detailed requirements-gathering through further in-country assessment.
2. Document system requirements, via use cases or other methodology, in the following domains:
    - Type of data handled in MB system, data collection and data entry process
    - Data management and quality assurance
    - Reporting and data analysis processes and functions
    - Role of paper-based information systems
    - System architecture, hardware and software performance
    - Interaction with electronic information systems used for other purposes in the health domain, including common data exchange formats
3. Document process requirements, as appropriate in the following domains:

- Participatory design
- Usability
- Development/acquisition/configuration
- Testing
- Deployment
- Maintenance and support
- Curriculum development and training

Activity Description

I-TECH will conduct a follow-up 3- week visit to Mozambique in early 2009 to gather further information on detailed requirements for a revised MB system, taking into account stakeholder feedback on Phase I recommendations. Prior to the visit, I-TECH will develop tools and methods to support further information-gathering and to facilitate consensus-based decision-making by key stakeholders about detailed requirements. The visit will include additional district-level site visits to clarify detailed requirements. I-TECH will communicate closely with MOH and other stakeholders to understand the desired format for presentation of the requirements (RFP or other format). Following the second visit, I-TECH will complete documentation of the detailed requirements and will facilitate a half-day videoconference to review the documentation with MOH and stakeholders.

Deliverables

- Detailed requirements suitable to guide redevelopment/update of MB
- RFP to bid out the work/recommendations

Goal:	Mozambique MOH and stakeholder's vision for updated Modulo Basico (MB) system is clarified through documentation of detailed system requirements.	Timeline: Months											
		10	11	12	1	2	3	4	5	6	7	8	9
Activity	PHASE 1: INITIAL ASSESSMENT												
1.1	Review existing background documents on MB system and copy of exisitng MB Access database	X											
1.2	Develop interview guides, tools for rapid assessment of existing MB system and unmet health information system needs												
1.3	Visit 1 (2 weeks total): Conduct key informant interviews with MOH and observe system use at national level		X										
1.4	Visit 1: Meet with system developer to clarify database architecture of exsiting MB		X										
1.5	Visit 1: Conduct key informant interviews of district-level users and to observe data flow processes and performance in 2 sites		X										
1.6	Visit 1: Inventory, collect all existing documentation for data management processes related to MB		X										
1.7	Draft description of existing MB database architecture and validate the description with the system developer			X									
1.8	Complete Phase I report documenting findings and recommendations			X	X								
1.9	Review findings with MOH and stakeholders via interactive videoconference debriefing meeting				X								
Activity	PHASE 2: DETAILED REQUIREMENTS DOCUMENTATION												
1.1	Develop methodology and tools for detailed-level requirements gathering				X								
1.2	Visit 2 (3 weeks total): Conduct key informant interviews, observations, and site visits as needed to complete detailed-level requirements gathering				X	X							
1.3	Maintain regular communication with MOH and stakeholders to gain consensus on desired strategy for updating MB					X							
1.4	Complete detailed documentation of requirements in agreed format (RFP or development plan) and review via interactive videoconference debriefing meeting						X						

Appendix 2 – I-TECH Capacity in Public Health Informatics

I-TECH
INTERNATIONAL TRAINING & EDUCATION CENTER ON HIV

I-TECH Initiatives to Strengthen Health Information Systems

Strengthening patient health records and facility-based information systems

- In Haiti, I-TECH began developing an HIV electronic medical record (EMR) system, called iSanté, in 2005 in partnership with faculty from the UW's Clinical Informatics Research Group, the Haiti Ministry of Public Health and Population, and CDC GAP. The HIV EMR began with implementation of standardized paper records, based upon national treatment guidelines and input from Ministry officials and other stakeholders. Web-based retrospective batch data entry was implemented, followed by deployment on local servers at some sites, with migration to a point-of-care data entry interface. Patient data are replicated automatically between systems to support an integrated view of care delivery, and to provide off-site backup. As of July 2008, iSantéwas functional at more than 30 clinics, capturing longitudinal records for more than 22,000 patients.
- In Haiti and Côte d'Ivoire, I-TECH is supporting implementation of electronic laboratory information systems (LIS). In Haiti, in collaboration with a national LIS working group I-TECH completed an assessment of data management processes and needs in laboratories of various levels in the public and private sectors in Haiti. Next, I-TECH collaborated with the LIS working group to develop, pilot, and disseminate improved standardized paper-based registries and tools for tracking specimens and results. Then, I-TECH developed a Request for Information on existing electronic LIS systems that might be adapted for use in Haiti. The group decided to pursue adaptation for Haiti of OpenELIS, an open-source system developed by the Minnesota public health reference lab and modified for implementation in Viet Nam, for Haiti. I-TECH has identified core programming tasks to make the system applicable for both the national reference lab (LNSP) and other clinical labs and will support implementation of this system in Haiti in late 2008-early 2009. In Côte d'Ivoire, I-TECH is further adapting and improving OpeELIS for implementation at 3 laboratories.
- In Malawi, I-TECH placed a senior Monitoring and Evaluation Technical Advisor at the Lighthouse ART Clinic at Kamuzu Central Hospital in Lilongwe, from 2006-08. The I-TECH Advisor has collaborated with Baobab Health Partnership to refine an innovative point-of-care electronic data system for clinical care, with user-friendly touchscreen interface. This system is successfully used at the Lighthouse Clinic as well as in several district hospitals and captures data for approximately one-sixth of all patients enrolled in Malawi's national ART program.
- In Ethiopia, I-TECH provides technical assistance on scale-up of the national ART program in the Afar, Amhara, and Tigray regions. Working in close partnership with the Regional Health Bureau offices, the local HIV/AIDS Prevention and Control Offices, and the Ministry of Health, I-TECH provides a broad array of technical, clinical, and operational support to 37 hospitals and health

centers. As part of this effort, I-TECH has hired, trained and deployed more than 60 data officers at these facilities, all responsible for managing patient data and reporting on standardized indicators. I-TECH also supports 13 field-basedmentoring teams.The teams spend 4-5 days/month at each facility to reinforce quality of care and sound data management practices. This mentoring support has resulted in dramatic improvements in management of patient information at the facility level. For example, when I-TECH began working at Mekelle Hospital, there were 2 years' worth of ART data that had not been entered in registers, and over 50 duplicate "unique" ART numbers for Mekelle patients. Medical records were not filed in a way that accessing records was efficient, and it could take up to 2 hours to find a patient's card. I-TECH worked with implementing partners Johns Hopkins University and Tulane University to deploy a team which worked for 8 weeks to resolve this backlog of patient data by abstracting of 2 years of patients data into follow up charts, and entering of indicators into registers. The team worked closely with site-level personnel, so that high quality practices for managing patient data would be maintained.

Strengthening information systems for national monitoring and evaluation

- In Haiti, I-TECH is developing an automated interface between iSanté, the national HIV EMR system, and the national data system for HIV case reporting, surveillance, and program monitoring and evaluation.
- In Malawi, the I-TECH Advisor has supported the Ministry of Health to develop national registers and other tools for managing data on PMTCT, TB/HIV referral, and ART programs.
- In Ethiopia, I-TECH's efforts at the facility-level have resulted in dramatic improvements in data management in the 3 regions where we work. Specifically, I-TECH's support helped improve the number of ART facilities which submitted monthly ART program monitoring and evaluation reports , from 14 of 29 facilities (48%) in March 2007 to 36 of 37 facilities (97%) in September 2008. This improved reporting at the facility level translates into more reliable and valid data at the regional and national level.
- In Mozambique, I-TECH has initiated an evaluation of the country's electronic information system, called Modulo Basico, which reports aggregate facility statistics for HIV and other diseases to the district, regional and national levels. The goal is to identify requirements for re-development of this system, to support greater local understanding and ownership of the code base, improved data quality and timeliness, and greater capacity for analysis, visualization, and utilization of program data.

Strengthening human capacity in public health informatics

- In 2007, I-TECH began a mentoring program for District M&E Officers in Botswana. As a part of this program I-TECH collaborates with the Ministry of Local Government and other partners in providing the M&E Officers with in-service training and on-site support to improve the quality of on statistics from health facilities and community-based HIV programs.
- In 2009, I-TECH will launch an effort in Kenya to develop capacity of local informatics resources groups to manage and maintain electronic health information systems. This will involve developing capacity for engagement in open-source development communities for local adaptation of health information applications to meet specific local need.
- In 2009, I-TECH is launching a Fellowship program for monitoring and evaluation of the national ART program, in collaboration with US CDC Global AIDS Program and the Malawi Ministry of Health. I-TECH will place 3 Fellows within the Central Monitoring and Evaluation Division (CMED) and the Department of HIV/AIDS of the MOH. The fellows include 1 Monitoring and Evaluation

Fellow and 2 Information Technology Fellows. The fellowship program offers opportunities for mentoring, coaching, and applied training in strategic information to exceptional health care professionals in Malawi. The program includes individualized skills assessment and robust professional development opportunities.

Appendix 3 – Forms used for Modulo Basico Data Collection

Appendix 4 – Agenda for ITECH visit, Oct 20-28, 2009

Monday 20th October		
p.m.	ITECH Team arrives, transfers to hotel	
Tuesday, 21st October		
9 – 11 a.m.	Introduction and Logistics Meeting at ITECH	I-TECH Mozambique staff, ITECH UW (Bill, Christina)
Findings	The trip was initiated with a discussion focused on detailing the status of the current implementation, and the management structure overseeing monitoring and evaluation system. Marla and Joan introduced us to the managerial structure within MOH and CDC. We also discussed the upcoming agenda, the background of ITECH's presence in Mozambique, as well as CIRG's work internationally.	
12:30 – 15:00	Lunch In-briefing and orientation meeting at CDC	ITECH UW (Bill, Christina) CDC Moz (Mindy, Cate, Janise)
Findings	Mindy discussed in more detail the structure and nature of the relationship between CDC and MOH and the structure of vertical programs within MOH that address certain programs. She also discussed Rik (the primary developers) expertise and involvement in the development of Modulo Basico, and the uncertain plans for the future of the MB staff. Lastly, the trip plan was reviewed and revised slightly.	
Wednesday, 22nd October		
8:30 – 10:30	In-Briefings/Introductory meetings at Ministry of Health (may be a series of meetings) Objectives: *Review objectives of visit *Meet key stakeholders *Input on overall vision and direction of MOH Health Information Systems	MOH • Dra. Almeida • Dr Campione • Rik • DPC (DraGertrudes) • Others? ITECH UW (Bill, Christina) CDC Moz (Mindy, Janise)
Findings	Ercilia discussed the deficiencies of Modulo Basico and the importance of visualization tools to facilitate the utility of the data stored by modulo basico. Rik introduced the system briefly and the area in which it covers. Ercilia also discussed the movement of the responsibility of the program to a new department and her resignation. Bill and Christina discussed the plans for the project and background in HIS.	
10:30 – 12:00	Demonstration of Modulo Basico	ITECH UW (Bill, Christina) CDC Moz (Mindy, Janise) MOH (Rik, others?)
12:00 – 13:30	Lunch (1908 Restaurant)	
13:30 – 16:00	Key Aspects of Modulo Basico	MOH (Rik) ITECH UW (Bill, Christina) CDC Moz(Janise)
Findings	Rik demoed the national installation of MB. He reviewed with us the installation procedure, the code base, and database architecture of MB. He also reviewed the existing documentation and user training process. Rik	

	reviewed the resources that manage MB in country, including the staff and resources available within the provinces and districts.	
Thursday, 23rd October		
8:00 – 11:00	Meeting with UEM/OASIS Human Resources Constraints and Opportunities in Informatics in Mozambique	 ITECH UW (Bill, Christina) CDC Moz (Mindy, Janise)
Findings	Chris Seebregts, Carl , and Leopoldo described OASIS's work in Mozambique. They described in detail Leonardo's work and the trial implementation of DHIS in Mozambique. They discussed key reasons for its failure. They also discussed their work in building in country development capacity in Mozambique through curriculum development with UEM. Options for future consultation or support regarding the development and expansion of MB were discussed.	
13:00 – 14:30	Lunch	
14:30	Depart for XaiXai/Gaza	
Friday, 24th October		
8:00 – 10:00	Meeting with DPS/Gaza: Modulo Basico at Provincial Level	MOH (Rik, others?) ITECH UW (Bill, Christina) CDC Moz (Mindy, Janise)
Findings	Discussed the use of the system withint he provincial directorate, case surveillance is used fairly frequently. A tour of the computer room for data entry of MB, and introduction to the data entry staff and the IT specialist for the province. He mentioned communication and training of the IT specialists would be useful. We viewed and discussed user experience with modulo basico.	
11:00 – 13:00	Meeting with DDS/TBD in Gaza: Modulo Basico at District Level	MOH (Rik, others?) ITECH UW (Bill, Christina) CDC Moz (Mindy, Janise)
Findings	Discussed the use of the system at the district level, the paper system, how data is transferred from the health centers to the district, and then on to the provincial directorate. We also discussed with the medical director what would be useful within the system for clinical decision support and program management at the district and health center level. We met the data entry clerks for the district and took a tour of the facility for data entry and storage. We viewed the Columbia university HIV patient management application and reviewed how data from the system is transferred to MB.	
14:00	Return to Maputo	
Saturday, 25th October		
9:00 – 11:00	Initial Out-briefing and Next Steps	ITECH UW (Bill, Christina) CDC Moz (Mindy, Janise)
p.m.	Bill depart	
Monday 27th October		
All day	Further meetings on Modulo Basico, follow-up within MOH as needed	MOH (Rik, Celia) ITECH UW (Cristina) CDC Moz (Mindy, Janise)
Findings	Outbriefing to MOH detailing our preliminary findings and discussing any priorities or concerns that MOH had with our suggested workplan. Janise and Christina spoke with Rik, detailing their work in other countries	

	and discussing the necessity for improvement and incremental redevelopment of MB	
Tuesday 28th October		
Morning	Further meetings with Daniel Lee, to discuss TARV in MB	MOH (Rik, others?) ITECH UW (Cristina)
	Daniel discussed gaps in data quality at all levels. He expressed the need for standardized forms of communication between levels, specifically the district, provincial and national levels. He also discussed double entry and poor processes for data review and quality assessment . Joan and Christina mentioned some feature additions to MB that were either mission critical or if added would be useful to assure quality and completeness of data: • Validations with an option to note • A mechanism to export all reports • More complete and easily readable training/reference material • A mechanism for creating custom standardized reports Daniel had little faith in the validity of the data, because the practices for collecting these data are not standardized at the hospital center (e.g., some write in calendar books, some have registers which are not conducive to allow accurate and complete storage of the data necessary to report) level	
11:00 – 12:00	Outbrief/Wrap-Up with CDC and ITECH	ITECH UW (Christina) CDC Moz (Mindy, Janise)
Findings	Christina reviewed preliminary findings, and presentation and feedback from MOH out-briefing. CDC, ITECH and Christina all discussed next steps for phase 1 completion and preliminary phase 2 objectives.	
p.m.	Christina Depart	

Appendix 5 – Documentation

A. Installation Instructions

Installation procedure MóduloBásico SIS

1. Copy HisInst1 folder from flash to C:
2. Create new folder C:\HisVb
3. Run c:\hisinst1\Package2\setup.exe
4. As the setup starts, select CHANGE DIRECTORY, and select C:\HisVb as directory.
5. Complete setup process.
6. Copy HisVb folder from flash to C: (OVERWRITE)
7. Copy or Rename His1.mdb
8. Set Windows Display settings to 1024x768 and in 'advanced' choose DPI settings Large Size (120DPI)
9. Windows – Control Panel – Regional Options – Regional and Language Options – Customize – Decimal symbol = .and Digital grouping = ,
10. Same Settings make sure Date Format = "dd-MM-yyyy"
11. Set Excel Macro Security to 'Low' (Excel-Tools-Options-Security-Macro Security-Low)
12. Make shortcut on desktop for C:\HisVb\His1.exe
13. Run MóduloBásico SIS
14. (If required: Import Area Codes and Data)

B. User Manual

Manual doUsuário

Modulo Básico - Sistema de Informação para a Saúde – S.I.S.

Índice

Iniciar o Programa S.I.S. – Modulo Básico

Duplo-click no IconS.I.S. , no Windows Menu

I. . Menu do Programa Principal

- Clicar a barra de menu
- Use as setas para navegar pelas opções do menu
- *Todos Dados* é activado , por defeito, como menu inicial.

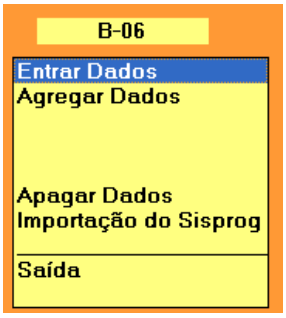
- Para PassarparaOutraFicha

  - Clique na Imagem da Ficha: A04 / B06 / B07 / B08 / C04 / D03 / D04

Exemplo B06

Composição do Menu do Programa Principal

- Secções do Menu



Teclas de Funções e Combinação de Teclas do Menu do Programa Principal

Alt-F4	Sair do Sistema -> Voltar ao Windows
Enter / Return	Aceder ao Programa Menu-Item

II. Gestão de Fichas

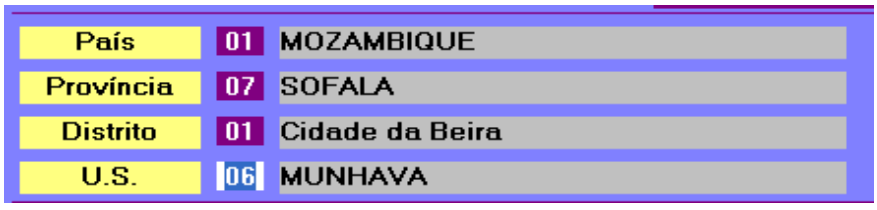
II.1. Entrada de Dados

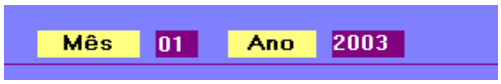
Entrada dos dados recolhidos manualmente, para as fichas mensais:

Método - Seleccione o Distrito
- Seleccione a Unidade
- Seleccione o Mês
- Seleccione o Ano
- Introduzaos Dados
- Grave a Ficha
- Saia

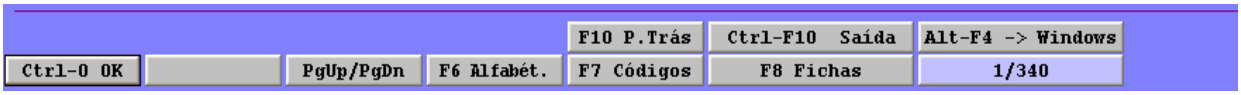
1.Seleccione o País, Província, Distrito, Unidade Sanitária

Campos:





- Teclas de Funções, Combinação de Teclas, ou Botões da Barra de Comandos:



Teclas Que Podem Ser Usadas Em qualquer Lugar do Ecrã:

F10	Sair do Campo –> Um Campo Para Cima ou Regressar Para o Menu Do Programa Principal.
Ctrl-F10	Ir Directamente para o Menu do Programa Principal
Alt-F4	Sair do Programa -> Ir Directamente Para O Windows
Setas	Movimentar Para Cima/Baixo através dos campos área/mês/ano
Enter	Ir para o campo Seguinte
Ctrl-O	Aceitar todos os códigos visualizados no ecrã e continuar com a ficha de entrada de dados. ! Manualmente para evitar ter que pressionar <Enter> em todos os campos do ecrã.

Seleccionar Código de área (País/Província/Distrito/Unidade Sanitária)

PgUp/PgDn	Navegar Para frente/traz através dos códigos de áreas
F6	Acesso ao sistema de pesquisa alfabética
F7	Listar códigos de área por código.
F8	Listar Fichas gravadas

F6 Aceder Ao Sistema de Pesquisa Alfabética:

- Janela adicional de pesquisa
- Escreva o nome ou os primeiros caracteres do nome do Distrito ou unidade sanitária pretendida, seguido de <Enter>
- Janela com listaAlfabética
- Navegue pela lista usando as setas ou o rato.
- Seleccione usando <Enter> ou duplo-click ou use F10 para sair da lista

F7 Listar Códigos de Área por Código:

- Janela com Listagem Numérica (por código de área).
- Navegar através da janela usando as setas ou o rato
- Seleccione usando <Enter> ou duplo-click ou, então, use F10 para sair da lista.

F8 ListarFichasGravadas

- Apresenta uma lista contendo todas as listas gravadas no sistema, para a área seleccionada, independentemente do mês/ano
- Navegue pela lista, usando as setas ou o rato.
- Seleccione a ficha existente, usando <Enter> ou duplo-click ou use F10 para sair da lista.

Seleccione os campos Mês/Ano

PgUp/PgDn Apresenta o próximo/anterior Mês/Ano.(Manualmente, para evitar que tenha que escrever os dígitos do mês/ano)

<Enter> No campo ANO: (igual a Ctrl-Oapartir de qualquer lugar do ecrã)
- Aparece a entrada de dados pelo ecrã.

C. Technical Documentation

Report Modules

1. Make Specific Exel Reports

T200.cmdboxEnter -> T200.Execution1 ->frmmod.Process2

Define reportlist (reports to be choosen from) in p1form.T301ScreenSpecific1
Set indicators enabled/disabled in p1form.CmdBoxItemEnable
Fill indicator list in p1form.Cmdbox1Lostfocus1
Finish selecting indicators and go to report in p1form.CheckExecution1

frmmod.Process2->prgcode="T301"
-> Before main loop calls FTodos1Rep.ReportAction1-> Calls P1Form.ReportAction1
P1Form.ReportAction1: - Opens RsReport1 (opens table LISTTodos1)
LISTTodos1: Ct, Pr, Dc, Hc, Yr, Mth, A1-A180
-> Main loop calls FTodos1Rep.ReportAction2-> Calls P1Form.ReportAction2
(only if index>0 (0=per indicator))
-> P1Form.ReportAction2: - Seeks + Addnew/Edit record in RsReport1
- Fill fields in RsReport1

-> After main loop calls FTodos1Rep.ReportAction3-> Calls P1Form.ReportAction3
-> P1Form.ReportAction3: - Copy OrigData/EmptyStat.xls to ….\Moz\Report\
- Open Exelwsheet
- Create OLE1 Link
- Define OLE1 size
- Fill ExelWsheet from RsReport1 file

frmmod.Process2 = general combine procedure

->P1Form.ReportAction1 = Open RSREPORT (on time or area) and zap it.
->Contbox
-> Loop
P1Form.ReportAction2
If report on diseases:
Set List(4).Visible = True, FlexList(3).Visible = False
Set T301A.OLE1
Call FrmMod.DisStat1(3, 2) 'Period Graph - 3=cmdboxindex for disease list
or
Call FrmMod.DisStat2(3) 'Area Graph
endloop

frmmod.DisStat1 = Periodical Statistic: Per Month/Quarter/Year or Total Period selected
setqrtr = week/month/trimester/year

seek existing, add record (write head) or edit record

Call P1Form.DisValue1 -> calculate value and write fields

RSREPORT1.Update
End Sub

P1Form.DisValue1

Call SelectDisease1(disnr, startfld, lines) 'look for the startfield position and lines (columns)
Case 1 '1st Cons.
startfld = 6

Call DisAdd(disnr, disam, startfld, lines) 'calculate values
Call DisAdd1 = add values
disam = disam + Val(RSFrm1Data(0)(startfld + ctr))

Call FrmMod.FillRepFileValue1(disam, 3) 'write values
RSREPORT1(disamfld) = Val("" & RSREPORT1(disamfld)) + disam
End Sub

Sub-Forms

Sub forms loaded on top of earlier loaded main forms.

Title Displays System Top Line and Selected Program title on top of screen.
Variables: mtit(2)
Mtit(0) = set in Start form, "Health Information System for Windows - Ministry of Health Zimbabwe - Version 1.0"
Mtit(1)= set in Start form, contains selected program title.

Mtit(2) not used (reserved)

Mess — Displays information message and command button bar at the bottom of the screen.

Variables:

- Command Buttons:
mg1(9) =>Contains Command options. Ex. F10=Exit
- Labels
PrgrLbl(1) =>Slider
mgLbl1 =>Percentage string on slider
message =>User info message content
- Textbox
mgtxt => located outside screen area, receives focus out of sight ex. in combine data

Procedures:

mg1_Click(sel As Integer) andmgtxt_KeyDown(KeyCode As Integer, shift As Integer)
* Check on Ctrl-F10 and Alt-F10 and call Main.ExitCheck if true.
* Set F10chk to true on F10

Fblank and Showmg16
* Clears all command buttons, sets selected number of records selected/total records
* Showmg16 sets specific selected number of records selected/total records
* Sets F10, Ctrl-F10 and Alt-F4 command buttons

Showmg(sel As Integer)
* Displays message variable (only, does not alter command buttons)
* If message longer then 45, mg6-mg8 hidden

PrgrLblOnOff(onoff As Boolean)
* If onoff hides command buttons and makes slider and percentage variables visible
* Slider and percentage calculated by ShowPrgrLbl

ShowPrgrLbl
* Calculates and refreshes slider and percentage

- Command Button Bar setting Procedures:

CbarArea()
CbarLocateFl()
CbarCode1()
CbarName1()
CbarFormEntry()
CbarBody1TbxField()
CbarBody1cmdbox1Field()
CbarBody1cmdbox1FieldA()
CbarList1()
CbarFormCombi()

CbarReport1()

- Command Button setting Procedures: (i As Integer)

Ok – CtrlS – PgDn – PgUp – Pgupdn - F2 - F4 - F5 - F6 - F7 - F7B - F8 - F8B - F9 - F10 - CF10 - AF4 – Pie – Bar – OleVb - OleEx

Area — Select Area level from combo-box.

Variables:

- Combo Box:
uarea => choices: National – Province – District - H.Centre - Group

Procedures:
uarea_KeyDown(KeyCode As Integer, shift As Integer)
* On Enter Calls Setarea(marea), SaveArea, CpdhSel.CpdhFocus
Setarea=Loads and display's all Area Codes and Names
Savearea stores selected area level in TSET
CpdhFocus sets focus to lowest area code level.

CpdhSel Area Code input.

Subforms: SFCPDH(4) as Code1, FlexList(0) as List1

Program Structure:

Public Sub LoadSFCPDH(Top As Integer)
* Called during form-load from T100, T200, T201, T400, T401
Initiates Code Form for Country up to Group with increase top position of 300 except for Group.
Sets tag for every code form.
Defines label text (Country, Province,…)

Public Sub Grid1Focus(KeyCode As Integer)
Displays F6 or F7 List
* Called from TbCode1Keydown (on F7), Crit1.TbxOnEnter
listtag = 0, List1.Visible = True, FlexList(0).Show vbModeless, HForm
Calls ShowFlexgrid1 depending on F6 or F7 and on itag for Country, Province, District, Health Centre
Ends calling List1.Grid1Focus
End Sub

Public Sub CpdhSel.TbCode1Keydown

F6 => CpdhSel.Crit1Focus
=> Crit.ShowvbModeless, HForm
Call Crit.TbxFocus(0) =>Crit.tbx(index).SetFocus

Crit.tbx_KeyDown
vbKeyDown Or vbKeyReturn
=>Crit.TbxOnEnter
=>CpdhSel.Grid1Focus(vbKeyF6)
=>seeF7

vbKeyF10 Or vbKeyEscape
=>Crit.Critexit
=>Crit.visible=False
=>CpdhSel.Critexit
=>SFCPDH(itag).TbCode1Focus

F7 => CpdhSel.Grid1Focus
=>Calls ShowFlexgrid1
=> Calls List1.Flexgrid1 index=0, string depending on F6/F7
=> Defines and Fills FlexList(0) (FlexList(0)=new List1)
=>callsList1.Grid1Focus
=>FlexList(0).Grid1.SetFocus
=>continues with List1.Grid1_KeyDown

F8 =>If Left(prgcode, 2) = "T4"

CpdhSel.Grid5Focus

Else CpdhSel.Grid3Focus

CpdhSel.Grid3Focus
=> Query on T5 and CT/PR/DC or HC file, open recordset
Calls List1.Flexgrid2 (itag=2)
=> Set FlexList(2).Data1.Recordset = mrecset
FlexList(2).Grid1.FormatString = frms
=> showsflexlist(2) (=List1)

Calls List1.Grid1Focus
=>FlexList(0).Grid1.SetFocus
=>continues with List1.Grid1_KeyDown

CpdhSel.Grid1Select

Listtag=0 (F6/F7)
=>CpdhSel.Grid1Select0 stores codes from flexgrid and reads/stores name fields into name labels

Listtag=2 (F8)
=> If NOT Left(prgcode, 2) = "T4"
=>CpdhSel.Grid1Select2 stores codes from flexgrid and reads/stores name fields into name labels

List1.Grid1Exit
=>CpdhSel.Grid1Exit
=> Call SFCPDH(itag).TbCode1Focus
(SFCPDH=Code1)

Code1 Area Code input.

Subforms: SFCPDH(4) as Code1, FlexList(0) as List1

Program Structure:

Public Sub LoadSFCPDH(Top As Integer)
* Called during form-load from T100, T200, T201, T400, T401

List1 => FlexGrid1, FlexGrid2,... filling and display of grids (lists on screen)
Grid1Focus, Grid2Focus,... sets focus to specific grid (list on screen)

Private Sub tbgrid1txt => sets value for tbgrid1 (= textbox variable) displaying the highlighted line in the list. Reads from list-matrix and stores into tbgrid1.

Exit and focus-set done in AForm- or HForm.Grid1Exit

Listtag=0: Grid1focus: Area Codes and Names list
->Grid1select->Grid1select0: accord area values

Listtag=1: Grid2focus: Group Area Codes List: only for F5 - T400
->Grid1select->Grid1select1:

Listtag=2: Grid3focus,Grid4focus,Grid5focus,Grid6focus
Grid3focus: F8 List Recorded T5 Forms + HCentre Names
Grid4focus: F8 List Recorded T5 Forms only codes
Grid5focus: F8 Recorded Population Records
Grid6focus: F8 Recorded Exported Population Records

List1.Grid1_KeyDown
vbKeyF10 or vbKeyEscape =>List1.Grid1Exit

vbKeyReturn =>AForm.Grid1Select
=>List1.Grid1Exit

List2 => F9 Screens List Menu

Variables:
- List Box:
listbox => choices: data-entry form screen titles

Public Variables

HFORM Form Variable. Value: Main Program Form Name
Called from general procedures to execute specific Main Program Action
Example: Hform=T100
Called from YesNo – Keydown procedure to execute T100.Cont1

Firstfocus Integer. Set to 1 on form-load. Used to exit tbx-focus at loading forms.
Set to 0 on tbx focus. If set to 0, tbx string checked on enter tbx value.

Init.bas Module Public Procedures

- InitForm Public Sub InitForm(mform As Form, Left As Integer, Top As Integer, Height As Integer, Width As Integer)

  Sets form – left, top, height,width, font = "Ms Sans Serif", FontName = "Ms Sans Serif", fontsize = 9, PaletteMode = 2, WindowState = 0

Areamod.bas Module Public Procedures

- Setarea Public Sub InitForm(mform As Form, Left As Integer, Top As Integer, Height As Integer, Width As Integer)