BID INITIATIVE

Better data. Better decisions. Better health.

PEOPLE PRODUCTS POLICIES PRACTICES

BID INITIATIVE BRIEFS: RECOMMENDATIONS AND LESSONS LEARNED

Software development cycle

The BID Initiative is grounded in the belief that better data, plus better decisions will lead to better health outcomes. It was designed in partnership with countries to enhance immunization and overall health service delivery by improving data collection, quality, and use. The BID Initiative takes a holistic approach to address immunization data challenges and strengthen evidenced-based decisions through a package of interventions including people, products, policies, and practices. These briefs summarize the approaches and interventions that the BID Initiative rolled out in partnership with the governments of Tanzania and Zambia and shares recommendations and lessons learned for others interested in improving immunization data quality and use.

BACKGROUND

The BID Initiative's software development cycle focused on the development and adaptation of electronic immunization registries (EIRs) to address key data quality and data use challenges. Many of the lessons learned with the software development cycle centered on greater and smarter involvement of key people in the process to ensure a product that will provide better access to information and the ability to build skills to improve data quality.

The national EIR was one of the most significant and intricate interventions developed to address critical data-related challenges identified by government officials and partners. The EIR aims to ensure that all children are registered from birth and do not miss a potentially lifesaving vaccine.

The creation of a national-level immunization registry requires an intense software development process. Successful development requires in-depth knowledge of routine immunization services, clinical care, and vaccine schedules; an understanding of the micro-level, individual-level data that will feed into the EIR, as well as the macro-level data on system outputs. It also requires knowledge about the all-encompassing way in which the data are used to improve immunization service delivery at each level of the health system.

For more details on registries, refer to our *EIR* brief.

The software development process can be divided into a number of standard steps:

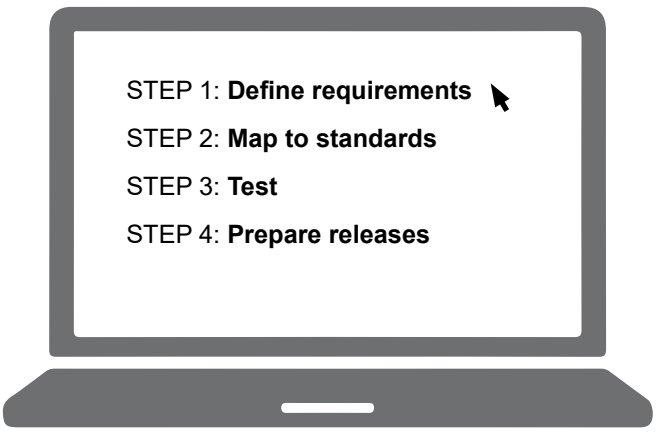
PATH

Types of system requirements

A functional requirement specifies what a system should do, such as the data elements to be collected and the reports to be produced. A non-functional requirement defines system attributes such as security, reliability, performance, maintainability, scalability, and usability.

- **Defining requirements.** The first step in the development of an EIR is to understand the current state of the challenges faced and the desired outcomes. This is done by collecting information on the critical requirements (functional and non-functional) for the information system. A comprehensive and early understanding of system requirements helps avoid many costly revisions later and identifies areas where interoperability with other information systems are critical. Both Tanzania and Zambia used the Collaborative Requirements Development Methodology (CRDM, developed by the Public Health Informatics Institute[1]) to collect and document functional and non-functional requirements for the EIR within the context of each country's program.
- **Mapping to standards.** Digital health solutions need to adhere to standards that can be categorized into the five C's: care guidelines, content, coding, communications, and confidentiality, privacy, and security. Many countries define the standards they want systems to follow, and it is important to align with any strategy or policy established by the government in this area.
- **Testing.** Test cases that reflect clinical workflow scenarios are mapped to the functional requirements and should be developed in collaboration with the software developers to ensure each round of testing is consistent and comprehensive. The testing should be conducted with several audiences in a variety of environments. These should include facilities, in the context of the clinical workflow; user advisory group members, who can advise on changes during each iteration; and different levels of the health system to ensure that all data needs are met.
- **Preparing releases.** Each release of the system as it goes through the development stages should have a unique release number and be published with release notes that capture which requirements are included in the release, the major changes made, and which bugs were addressed.

FIGURE 1: The Software Development Process



Types of standards

CARE GUIDELINES
These include evidenced-based protocols such as the immunization schedule recommended by the World Health Organization.[2]

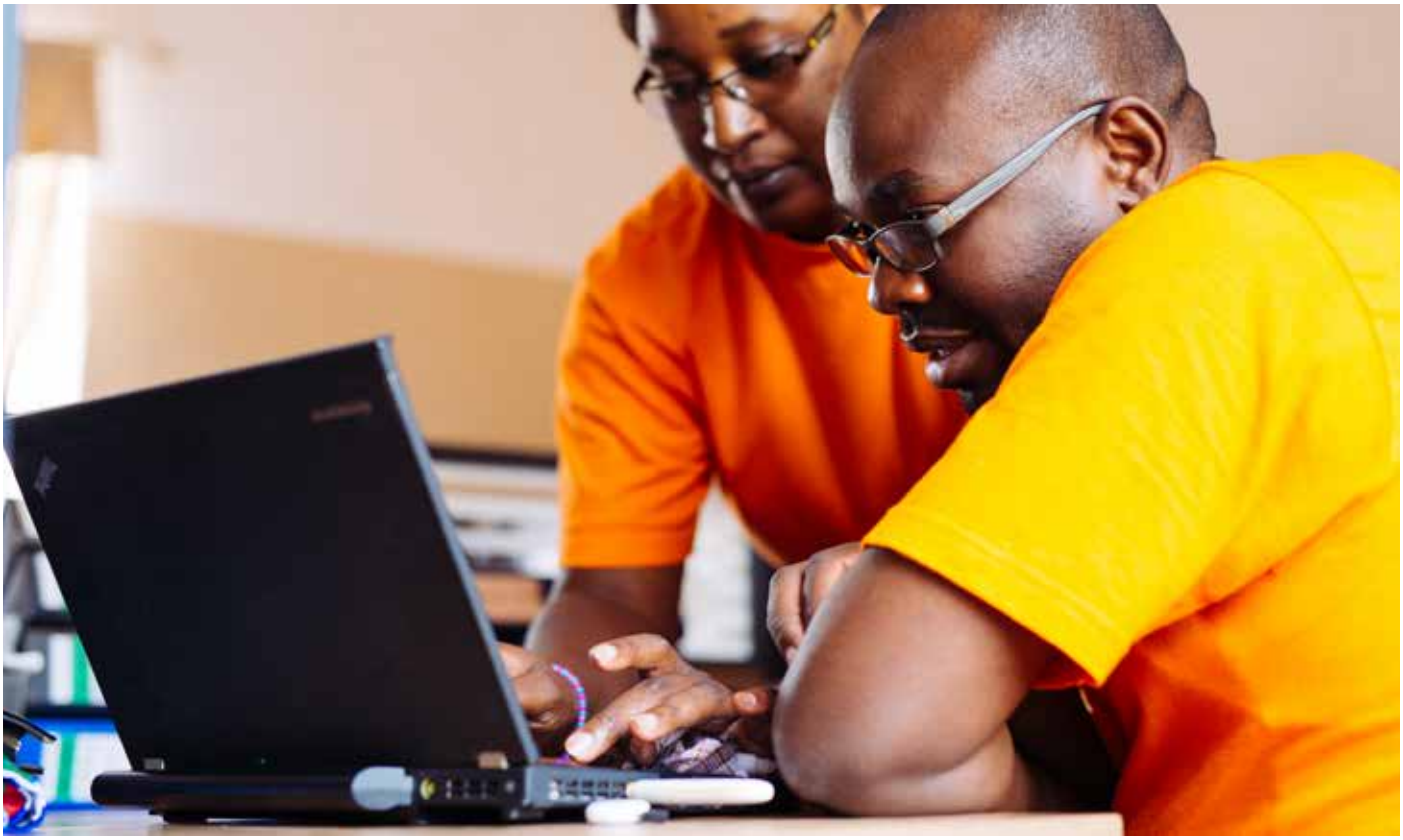
CONTENT
This can be thought of as the data fields on the paper form. Examples of content standards include HL7 clinical documentation architecture templates, such as the immunization content specification.[3]

CODING
These apply to specific fields on a paper form. A form field intended to capture values for "SEX," for example, could follow the ISO 5218 specification: 0=unknown, 1=male, 2=female, 9=not applicable.[4]

COMMUNICATIONS
For interoperability, it is important that the protocols around this messaging adhere to precise standards such as Integrating the Healthcare Enterprise's (IHE) cross-enterprise document-sharing specification (XDS).[5]

CONFIDENTIALITY, PRIVACY, AND SECURITY
Personal health information must be kept private. Standards include health care-specific profiles such as the IHE Basic Patient Privacy and Consents specification and cross-industry standards for authentication, encryption, and secure communications such as OAuth (open standard for authorization), public key infrastructure, and Transport Security Layer.[6]

1. CRDM, Public Health Informatics Institute. https://www.phii.org/crdm.
2. World Health Organization (WHO) recommendations for routine immunization - summary tables. WHO; 2017. Available at www.who.int/immunization/policy/immunization_tables/en/.
3. Integrating the Healthcare Enterprise (IHE). IHE PCC Technical Framework Supplement – Immunization Content. Oak Brook, IL: IHE; 2011. Available at www.ihe.net/Technical_Framework upload IHE_PCC_Suppl_Immunization_Content_Rev2- 2_TI_2011-09-09.pdf.
4. International Organization for Standardization/International Electrotechnical Commission websites. Available at www.iso.org and www.iee.ch.
5. Cross-Enterprise Document Sharing page. IHE website. Available at http://wiki.ihe.net/index.php?title=Cross-Enterprise_Document_Sharing.
6. Basic Patient Privacy Consents page. IHE website. Available at http://wiki.ihe.net/index.php?title=Basic_Patient_Privacy_Consents.



PATH/Gabe Bienczycki

RECOMMENDATIONS BASED ON LESSONS LEARNED

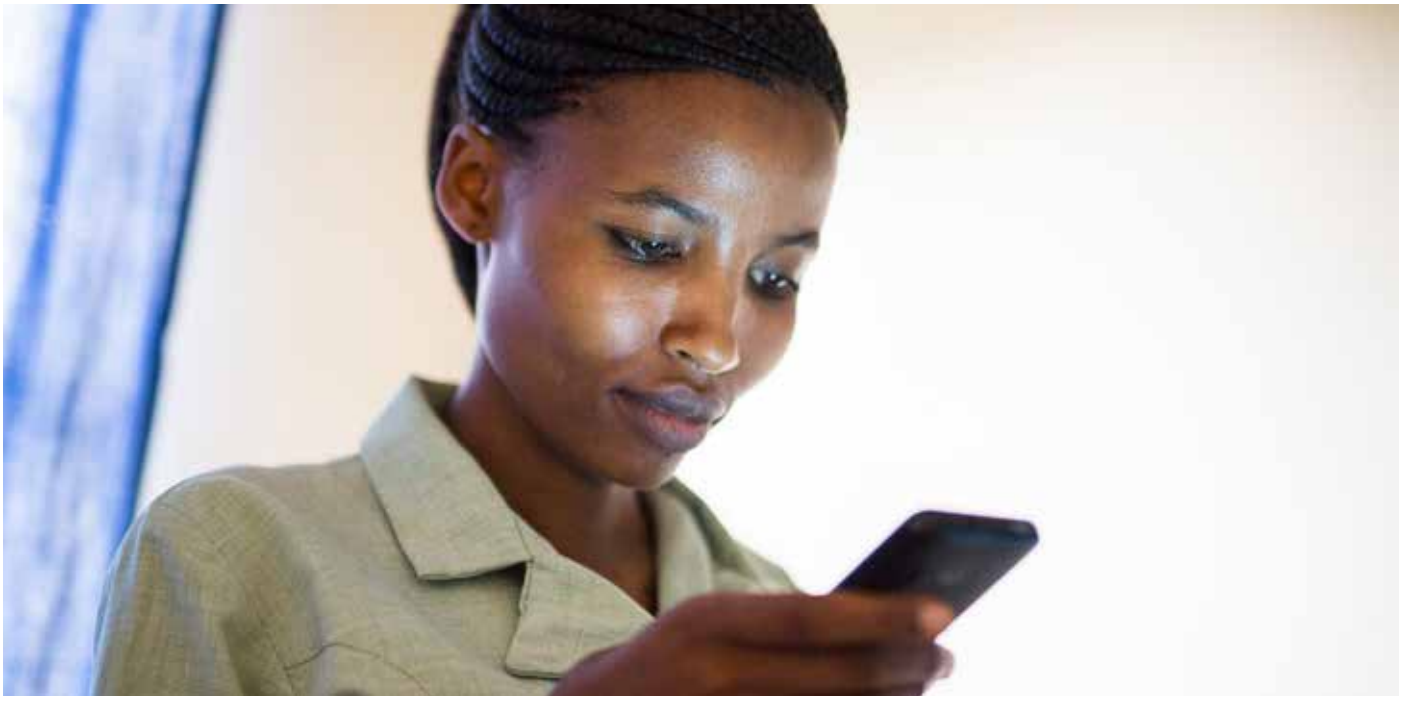
Several recommendations have emerged based on lessons learned during software development in partnership with the governments of Tanzania and Zambia:

1. **Identify a product owner.** The process requires someone to act as a product owner to work with the system developers. This person must understand the national vision for the system, as well as the more granular details of how the requirements are put into action in the system. He or she also understands the data flows (inputs and outputs) as well as the clinical context. The product owner is also responsible for directing the activities, working with the developers, and coordinating partners, so timelines and context are always understood and aligned.

2. **Set expectations for timelines and resources.** Based on the documentation of the functional and non-functional requirements, it's critical for the product owner and developers to talk through and document all the details of each requirement at the inception meeting. This process allows developers to understand the nuances of the functionality desired and helps the product owner understand the technical work needed. More importantly, this facilitates the creation of a realistic development timeline that allows time for iterations, as well as the allocation of appropriate resources to deliver releases on time and to meet expectations.

3. **Design devices and apps for multi-user environments where data are shared offline between users.** Nurses in a facility typically rotate from one care service to another, which requires functionality that supports the log-in and data entry of multiple unique users on the same device. We found through the process of several software development cycles that this use case is very demanding on software architecture, especially in facilities that may often be offline. This made it difficult to find a software platform compatible with tablets that would allow for offline functionality, enable multiple users to access and enter data into the same device, and allow for the production of an audit trail to see which nurse provided which service and when.

4. **Build realistic software development timelines.** Through the assessment of several software platforms that responded to requests for proposals, we found that many were not ready to go in their current state and would require additional development work. Most proposals suggested at least six months of development, and we even experienced longer times for some of the actual software development process for both countries. This not only created "learning costs" but also changed the landscape of several of these software platforms to understand what it would take to meet the forward-thinking needs of the countries. The development of EIRs in Tanzania and Zambia should significantly minimize learning costs for other countries looking to use these platforms to address their immunization-related data challenges.



PATH/Trevor Snapp

5. **Ensure key ministry of health (MOH) staff at a national level participate in the entire process.** Both countries had specific strategies for the facility, district, and provincial/regional levels. It's also important to have a strategy for the national level. Participation from experts at the MOH in clinical care, information and communications technology, monitoring and evaluation, and immunization allows the team to identify gaps at earlier stages and make decisions together as challenges arise.

6. **Employ enterprise mobility management (EMM).** Software and hardware require ongoing maintenance. EMM allows for the management of devices to see which versions of software applications are being run on the device. It also allows for devices to be wiped if they are stolen and helps to track the location of connected devices.

7. **Consider areas of potential scope creep and where it may be necessary to ensure a comprehensive view of the child's health.** It's often difficult to avoid scope creep or new requirements based on user feedback during the iterative process. For example, the EIR collects the mother's data but does not collect her health history. However, in the context of the child's routine immunizations and services, both EIRs included data on deworming and vitamin A. Tanzania's EIR also includes data on the tetanus protection of the child, which comes from the tetanus vaccines received by the mother. This was done both for the child's comprehensive record and to capture the data elements in the immunization section of the Health Management Information Systems.

For more details on our implementation approach, refer to our *rollout strategy* brief.

Cover photo: PATH/Trevor Snapp

BID Initiative

Street Address
PATH
2201 Westlake Avenue, Suite 200
Seattle, WA 98121 USA

Mailing Address
PATH
PO Box 900922
Seattle, WA 98109 USA

206.285.3500
www.bidinitiative.org
bidinitiative@path.org

September 2017