Scaling the capacity for BLIS (Basic Laboratory Information System) Implementations across laboratories in Africa

Submitted by Emmanuel Kweyu (Strathmore University) on January 19, 2018 - 6:22am
Last revised by Web Producer on June 21, 2018 - 3:09pm.
Proposal Status: Postponing for Future Calls for Proposals

1.0 Executive Summary

Strathmore University requests support for the scaling of in-country technical and implementing capacity for the expanding open source BLIS (Basic laboratory Information System) user community across selected laboratories in Africa.

As a background Laboratories in developing countries, have a high demand for test services but are under equipped, understaffed and compounded by workflow inefficiencies. Strathmore University in collaboration with other implementing partners have developed an open source Basic Laboratory Information System (BLIS) that was configured primarily for specimen, testing and test results management supporting functionalities such as lab test equipment interfacing and electronic data transmission to other systems(e.g. EMRs) to reduce manual workload, decrease turn-around times, and improve quality control, and documentation to meet internationally-recognized laboratory standards and improve quality of laboratory testing.

BLIS has been successfully implemented in 2 county hospital laboratories in Kenya with ongoing implementations in 10 district/regional laboratories in Uganda. This implementation experience hand proof of concept has resulted in emerging interest and requests for technical support from a number of countries including Swaziland, Mozambique and Nigeria. We have also developed a growing community of BLIS users including developers, implementing partners, lab managers and technologists and the ministries of health stakeholders.

This proposal is a request to support the scaling and building of local in-country capacity to sustainably support BLIS implementation to laboratories in Africa. This objective will be achieved through the following key activities;

1) Developing and strengthening of the in-country technical capacity through the establishment of the "BLIS Academy" to focus of developing both offline and online training resources targeting various categories of the BLIS user community.

2) Strengthening the development of the BLIS core modules to support a wide ranges of workflows, services, tests and specimen referral support.

3) Support the BLIS integration to more laboratory testing equipment and data exchange with other systems.

4) Provide technical support to the BLIS user community and drive the expansion of the community through a well-defined communication strategy.

Consortium Team

The project will be led by the Strathmore University-BLIS Management Team. The software development will be implemented by the core BLIS developers and the implementation, testing and user training will be supported by the Association of Public Health Laboratories (APHL), the Ministries of Health at the National and County levels in respective countries and the Centers for Disease Control and Prevention (Kenya office).

The key contact person will Emmanuel Kweyu: ekweyu@strathmore.edu

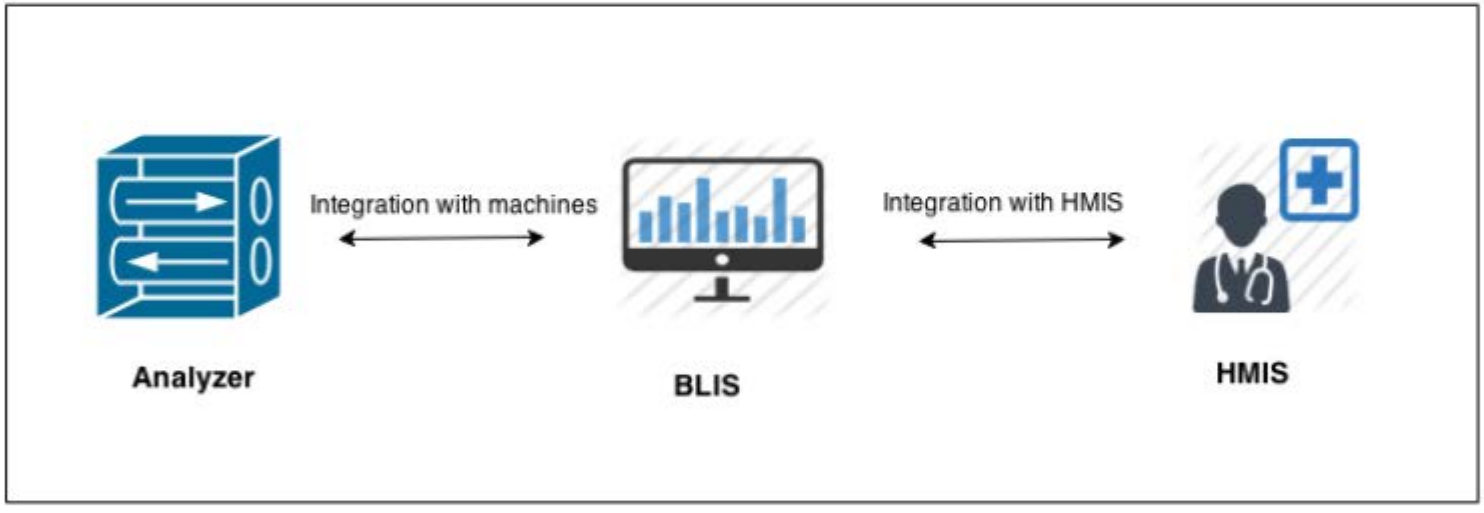
2.0 Detailed Project Proposal

The Basic Laboratory Information System (BLIS), is an open source Web-based system that can be installed in a local, district, or national laboratory. It is a tool that can help to standardize data, which improves the ability to run useful reports and can both give a realistic picture of laboratory services and assist with staff and budget planning. With enough data, BLIS can be used to track disease prevalence over time

Features of BLIS include:

- One-time entry of each unique patient
- Standardization of data collected (allowable entries for specimen type, test type, patient data, reagents
- are set at MOH level and then entered consistently throughout a country)
  - Customization to a country's needs
  - Ability to track lab supplies such as test kits, reagents
  - Ability to run reports as specified by a country
  - Automatic alerting of data values that may be out of range(reference ranges and panic values are set at the regional or national level

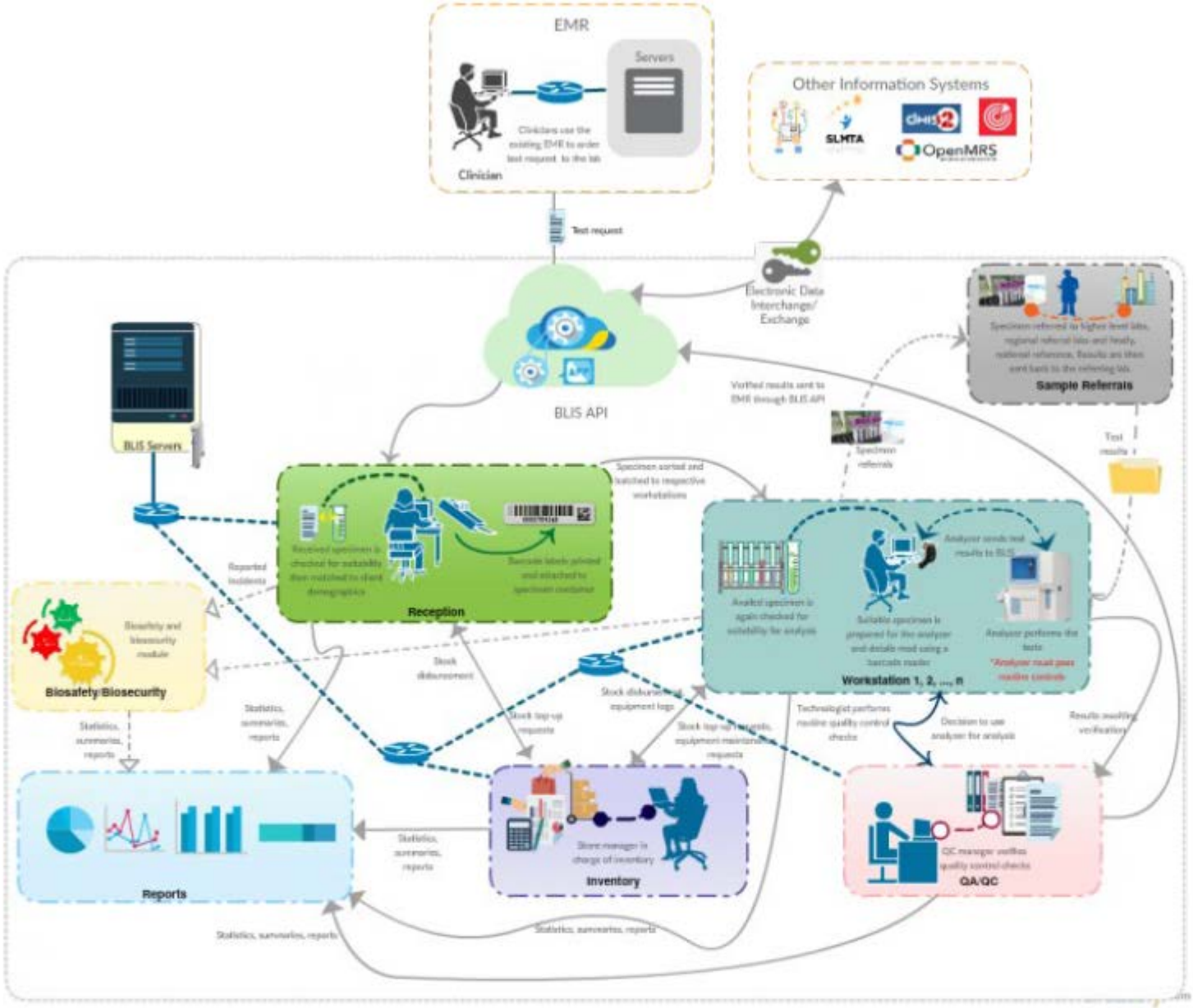
The BLIS Kenya projects is on Github . There is a simple guide on how to set up the system. We also have a technical wiki on how the system works internally, coding patterns to use and a general system walk through. An assortment of documentation with relation to lab instruments, SOP's and old documents can be found here . A schematic model of the BLIS system is illustrated below;



The proof of concept of the project has been successfully implemented in 2 laboratories on Kenya with current ongoing implementation in Uganda. The key issues for the project is build the technical capacity and expand the development and scaling of BLIS to more laboratories , support more services, and integration to instruments/equipment and other systems.

2.1 BLIS- Use Cases/ Stories and Activities.

The Scaling of the BLIS project is based on mapping and support the general workflow and use case that is used in all public medical laboratories in Africa. Typically the workflow and use cases is provided by the diagram below.



The general workflow has reception (specimen registration and processing), workstation (testing and results entry), reports, QA/QC, inventory, sample referrals, device automation and the API modules as the major components. The actors and processes of the core use cases are provided below

2.1.1 Use Case

Use Case	Recommend Tests
Actors	Clinician, External system

Description	The clinician orders certain tests for the patient whose information is held in paper forms or the external database, the client then goes ahead to pay for the tests then submits the receipt at the laboratory reception.
Data	Patient's personal information, diagnosis summary
Stimulus	User command issued by the clinician
Response	Confirmation that the records have been saved
Comments	Appropriate security permissions should be in place to enable the clinician perform the above duties.

Use Case	**Confirm Patient Details**
Actors	Laboratory receptionist, BLIS
Description	The laboratory receptionist confirms the patient details which will be imported from the external system or manually by checking against the payment receipt presented by the patient.
Data	Patient's personal information, diagnosis summary, recommended tests
Stimulus	User command issued by the laboratory receptionist
Response	Confirmation that the records are available
Comments	The laboratory receptionist should be only allowed to view information appropriate to their role.

Use Case	**Register Specimen**
Actors	Laboratory receptionist, BLIS
Description	Here, patient proceeds to give the required samples depending on the tests. The specimen are then registered and barcoded for confidentiality then sorted to the various benches as appropriate.
Data	Patient's personal information, tests prescribed, specimen details
Stimulus	User command issued by the receptionist
Response	Confirmation that the records have been saved
Comments	Appropriate security permissions should be in place to enable the receptionist perform the above duties.

Use Case	**Receive Specimen**
Actors	Bench receptionist, BLIS
Description	After sorting to the different benches, the bench receptionist picks the specimen for which they must acknowledge receipt and state whether they are viable or give reasons for rejection. The receptionist then schedules the tests for the specimen in that particular bench.
Data	Barcode, specimen
Stimulus	User command issued by the receptionist as regards the specimen and tests.
Response	Confirmation that the records have been saved
Comments	Appropriate security permissions should be in place for the bench receptionist to access the information necessary.

Use Case	**Schedule Tests**
Actors	Bench receptionist, BLIS
Description	The bench receptionist proceeds to queue the specimen that are viable in the system ready awaiting the technologist to perform the necessary tests.
Data	Specimen information, tests summary
Stimulus	User command issued by the receptionist as regards the specimen and tests.
Response	Confirmation that the records have been saved
Comments	Appropriate security permissions should be in place for the bench receptionist to access the information necessary.

Use Case	**Define Reference Ranges**
Actors	Laboratory quality control officer, BLIS
Description	The quality officer here should define the reference ranges for various reagents used in the different tests so that the system is able to automatically state the intensity of a test result.
Data	Reagent details, Results details
Stimulus	User command issued by the quality officer.

Response	Confirmation that the records have been saved
Comments	Appropriate security permissions should be in place for the quality officer to access the information necessary.
Use Case	**Update Reference Ranges**
Actors	Laboratory quality control officer, BLIS
Description	Normally, reagents differ and various testing machines have different ranges of normal values. Therefore, the quality officer should be updating this information frequently.
Data	Testing machines, reagents details
Stimulus	User command issued by the quality officer.
Response	Confirmation that the records have been saved
Comments	Appropriate security permissions should be in place for the quality officer to access the information necessary.
Use Case	**Update Test Results**
Actors	Laboratory technologist, BLIS
Description	The technologist, after performing the tests on the different specimen for the various patients, they proceed to enter the results into the system which will await confirmation by a different party.
Data	Tests summary, Results details
Stimulus	User command issued by the lab technologist.
Response	Confirmation that the records have been saved
Comments	Appropriate security permissions should be in place for the technologist to access the information necessary.
Use Case	**Verify Test Results**
Actors	Laboratory technologist/Lab manager, BLIS

Description	After the results are entered into the system, they are not released yet to the clinician, they should be cross checked for verification by a different party from the entering party.
Data	Tests summary, Results details
Stimulus	User command issued by the lab technologist.
Response	Confirmation that the records have been saved
Comments	Appropriate security permissions should be in place for the verifying party to access the information necessary.

Use Case	**Results Interpretation**
Actors	Clinician, external system
Description	Having been verified, the test results are interfaced with the external system so that the clinician is able to interpret them to determine the appropriate medication.
Data	Tests summary, Results details
Stimulus	User command issued by the lab technologist.
Response	Confirmation that the records have been verified
Comments	Appropriate security permissions should be in place for the clinician to access the information necessary.

Use Case	**Update Inventory**
Actors	Bench receptionist, BLIS
Description	The bench receptionist has the duty of updating the level of inventory so that the reorder level from the stores is known and well managed. This is a frequent exercise.
Data	Reagents details, stock levels
Stimulus	User command issued by the receptionist.
Response	Confirmation that the records have been saved
Comments	Appropriate security permissions should be in place for the bench receptionist to access the information necessary.

Use Case	**Perform Control Checks**
Actors	Laboratory quality control officer, BLIS
Description	The quality officer is tasked to frequently check the various machines and operating procedures used at each bench. This shall ensure that any unusual system behavior is noted earlier enough.
Data	Tests summary, Results details, reagents details
Stimulus	User command issued by the quality manager.
Response	Confirmation that the records have been saved
Comments	Appropriate security permissions should be in place for the quality manager to access the information necessary.

The detailed core functionalities are provided in the Main appendix document.

2.2 Digital Health Technologies- BLIS Core Development Technologies

The BLIS project Implementation is based on the General Public Open source Tools and Technologies. The source code and associated development tools and open and available to the BLIS community. This is an importance component of collaboration and building of technical capacity in respective country communities.

The table below provides the tools and technologies stack used in the development and deployment of the BLIS project.

Development Component	Tools/Technologies
Front-end Design	**AngularJS,CSS,HTML5**
Backend design	**Php/Java MVC Frameworks:** **Laravel.** **Spring boot.**
Database	**MySQL, Postgres, MSSQL**
Modelling/Prototyping Tools	**Draw.io, Microsoft Visio**
Penetration Testing Tools	**Kali Linux Tools**
Network Security	**IPSEC, OpenVPN**
Project management	**MS Project,**
Cloud tools	**Nginx,Apache, Amazon web services**

Code Hosting	Bitbucket
Continuous Integration tools	Travis CI
Project Source Code Repository	GitHub
Online Training and Support Materials	Moodle- Learning Management System

For operational efficiency, we provide below the minimal hardware infrastructure to deploy the system in a typical high capacity laboratory serving approximately 10,000 tests per-month.

2.2.1 Hardware

- 1x Dual-Core Server-CPU
- 8 GB working storage (RAM) (at least 2GB RAM)
- 500 GB free hard disc capacity
- Network TCP/IP, 1000 Mbit
- Graphic resolution 1024 x 768 pixels, 16-bit colors
- Keyboard, Mouse
- Standard Smart Phones: Minimum OS - Android Lollipop 5.0 and above

2.3 Community Feedback Methodology

The Scaling of the BLIS project will require as part of long term objective of sustainability and retaining local ownership an Active Participatory Approach involving all the stakeholders. The key stakeholders in the Scaling are the BLIS Community comprising of the core developers, implementers, lab users and the hospital/country administration and partners.

The community engagement will be based on the Agile Systems Development Methodology. Agile Development is an umbrella term for several iterative and incremental software development methodologies. Agile development methods are characterized by the short period generally 1-4 weeks in which software is developed. These periods are called iterations and can be considered sub projects that deliver something operational at the end of the period.

Prototypes are tested and assessed, after which improvements can be implemented. In this way, the development team knows whether it is heading in the right direction and a new iteration can be started without any risk. Not only is the product assessed, but also the development process. The goal here is to strive toward continuous improvement. Another characteristic is that agile is a method that produces a sub product that actually functions after each iteration, which can be immediately incorporated to an existing software to test whether everything works well.

The table below provides summary of the method of engagement of the BLIS community that will provide a basis to scale the project across Africa.

BLIS Community Stakeholders	Method/Tool of Engagement	Frequency of Engagment
Core BLIS Developer Community	• Face to face Trading and capacity Building workshops • Online Skype Calls • Online Slack Channels for managing the development and Testing of the code • Email Mailing Lists for Announcements and progress reporting • Online eLearning Platform for Training and Learning Resources • Face-to-face – 4days(3 times a years) • Weekly/Monthly Skype calls • Online platforms will be continuously available	
BLIS Implementation and Training Community	• Face to face Trading and capacity Building workshops • Online Skype Calls • Online Slack Channels for managing the development and Testing of the code • Email Mailing Lists for Announcements and progress reporting • Online eLearning Platform for Training and Learning Resources • Face-to-face – 5days(5 times a years) • Weekly/Monthly Skype calls • Online platforms will be continuously available	
BLIS Users (Lab Technicians/Quality Officers etc)	• On- site Training and user Support • Joint Group workshops in selected countries • Online help desk for support • Email Mailing Lists for Announcements and FAQs responses • Online eLearning Platform for Training and Learning Resources • Onsite training -3-5days • Joint workshops per country(2-3 times per year) • Online platforms will be continuously available	
Hospital Administration	• Joint Group workshops in selected countries • Email Mailing Lists for Announcements and FAQs responses	• Joint workshops per country(2-3 times per year) • Online platforms will be continuously available
Ministries of Health/ Donor Partners	• Joint Group workshops in selected countries • Email Mailing Lists for Announcements and FAQs responses	• Joint workshops per country(2-3 times per year) • Online platforms will be continuously available

2.4 BLIS Self-Assessment of Global Good Maturity Model

The table below provide the summative self-assesment of the BLIS system Global Good Maturity Model. The scaling of the project with input from the stakeholders across Africa **BLIS Rating Rating of a Digital Health Software Global Good (make a copy of this document to use)**

Core Indicator and Calculated Score [0-10]	**Sub-Indicator**	*change rating here*	
Global Utility	Country Utilization	Low	Less than two countries or states actively use the tool for use as part of their health information system
6	Country Strategy	High	At least ten countries or states have included the tool as part of their eHealth strategy or framework
	Digital Health Interventions	Medium	the tool does partially meets digital functional requirements (as defined by WHO's Classification of Digital Health Interventions) without signifigant customization or configuration
	Source Code Accessibility	High	source code exists on a publicly accessible repository and licensed under an Open Source Initiative approved license. Software is structured to allow local customizations and new modules and functionality without requiring forking of main code
	Funding and Revenue	Medium	multiple revenue streams/funders exist across project implementations
Community	Developer, Contributor and Implementor Community Engagement	Low	Less than 10% of estimated total of developers, contributors and implementers are on a communication platform
3	Community Governance	Medium	some informal processes for community management exist to direct continued development of the digital health tool

	Software Roadmap	Medium	there is a publicly accessible and routintely maintained platform for new feature requests. a software roadmap exists describing currently planned and resourced development activities
	User Documentation	Medium	some user documentation exists (training manual, demo videos) but only addresses a limited subset of common functionality
	Multi-Lingual Support	Low	Limited or no support in the software for multiple languages. Multi-lingual documentation / user resources are practically non-existent
Software	Technical Documentation	Medium	some technical documentation exists of the source code, use cases and functional requirements
6	Software Productization	Medium	full documentation available for deployment and configuration. a new implementation does not require the involvement of the core development team
	Interoperability and Data Accessibility	Medium	some APIs are available for accessing and managing data. there are user facing interfaces to export core data and metadata in the system (e.g. in CSV format) for further analysis and data transfer purposes
	Security	High	Role based authorization exists, if appropriate. All remote access (web interface, APIs) are encrypted by default using current best practices. An independent security audit of the software has taken place within the last twelve months.
	Scalability	Medium	There is at least one jurisdicion (e.g. country, state) deployment for which 20% of all "entities" are managed within the software. There has been at least one evaluation of software performance / load testing

2.5Project Workplan, Deliverable and Schedule

Activity	Sub-tasks	Duration(Months)	Deliverable
Establishing of the BLIS Academy Resources	• Developing the Training Resources for all users • Setting up of the online-Training platform • Running of the Trainer of Trainer (ToT) • Engagement with stakeholders to create awareness of the academy	Month 1-3	• Training Resource materials • Online Learning management platform to host the training
User needs assessments for enhancement of BLIS core modules	• User requirements workshops and site assessment for new workflows, instruments and systems integration,	Month 1-2	• Detailed requirements specification mapping the common requirements across the countries
System Development and Customisation	• Based on the user requirement, core system development of the BLIS Systems	Month 2-8	• Tested prototypes of core modules
Beta testing , Bug fixing and Implementation	• Using the implementing partner to test, report and implement the system	Month 5-8	• Test plans and reporting test results use cases
User Training workshops	• Targeting core system developers • Lab Managers and Technologist • Ministry of Health Managers	Month 2-8	• Certifications • Training assessment and feedback reports
User community Support and expansion of the community	• Establishing a reporting help desk, mailing lists for bug fixing, user support and upgrade support	Month-6-24	• User Roadmap for implementation and feedback for support

2.6 Project Implementation Schedule

The Gantt chart is attached in the summary proposal document.

Supporting Documents:

BLIS Concept Note

scaling_the_capacity_for_blis_implementation_across_public_health_labs_in_africa_main_proposal.pdf