iLab Africa

BASIC LABORATORY INFORMATION SYSTEM DOCUMENTATION

(BLIS)

Table of Contents

Table of Figures

HOME

1.0 Feature List (Current)

Authentication and authorization

- Login
- Logout

1. User Management
    - Create new user
    - View list of existing users
    - View individual user details
    - Update user details
    - Delete user

2. Patient Records

    - Create new patient
    - View list of existing patients
    - View individual patient details (personal information)
    - Update patient details
    - Delete patient
3. Lab Sections - Create, Read, Update and Delete
    - Create new lab section
    - View list of existing lab sections
    - View individual lab section details
    - Update lab section details

4. Test Types
    - Create new test type
    - View list of existing test types

- View individual test type details
- Update test type details
- Delete test type

5. Measures
   - Create new measure
   - View list of existing measures
   - View individual measure details
   - Update measure details
   - Delete measure
6. Specimen Types - Create, Read, Update and Delete
   - Create new specimen type
   - View list of existing specimen types
   - View individual specimen type details
   - Update specimen type details
   - Delete specimen type

1.2 Standards

1.2.1 General programming

All indices should zero based. e.g. if a **<select>** element has sequential numeric values the would be from the series

1.2.2 Database

Naming table names, column names: all lowercase, use underscores to separate words e.g **specimen_id**, **test_type_id** names with abbreviations may be treated differently e.g. **targetTAT** makes more sense • better long names that makes sense

- Use of appropriate field type e.g. '**SMALLINT**' instead of '**STRING**' for age If a primary ke is necessary on a table it should be the first column and should be named '**id**'. It should be unsigned.
- When this id is referenced in another table (as a foreign key) it should be called '**tablename_id**'
- Table names are in the plural e.g. **specimens** instead of **specimen**
- Create a migration file for every new table or change to a table. If necessary add the seed data using a separate file. See the Migrations and Seeding section of the online Laravel guide.

1.3 PHP Classes

This applies to Controllers, Models,migration and seeding classes;

1. Class names always begins with uppercase e.g. **CreateTestTypeMeasureTable**

2. When combining words every word begins with uppercase. Do not use underscore e.g **SpecimenType** as opposed to **Specimen_type**

3. Method or Function names begin in lowercase and words join in uppercase e.g. **getSpecimenTypes()**

4. Provide descriptive class and function comments before their definitions

5. Variable names – begin in lowercase. Multiple words should begin in upper case e.g. **specimenType**

6. Better long names that makes sense e.g. **specimenType** instead of **speType**

7. Soft delete(s) preferable to hard delete(s)

1.4 Laravel views

1. Using blade templating engine

2. All html elements in lowercase

3. Using bootstrap for css

1.5 HTML and CSS

1. All CSS should be placed in an external stylesheet file (public/css/)

2. CSS class names should use hyphens when joining words e.g ‘**user-image**’

3. Indent for readability in order of precedence e.g.

```
.form-group{}
.form-group.actions-row{}
    .form-group > label{}
    .form-group .form-control{}
    .form-group .form-pane{}
        .form-group .form-pane label{}
```

1. Never use !**important**;
2. CSS files should have categorized sections e.g. for user files, for general layout
3. Usecontextual elements such as <**strong**> and <**em**> instead of <**b**> or <**i**>
4. Page layout should be done from the CSS file. Do not use **&nbsp;** for spacing. If done properly there should be no need to use <**br**> either.
5. Comments: Include the section name e.g.

```
/**
 * forms
 */
```

1.5 Javascript

Using jquery

1.5.1 Github

1. Create an issue for each change you want to make.
2. Create a branch from the master for the same change. The branch name should be descriptive
3. Once fixed, create a pull request asking a specific person to review the feature.
4. Reviewed and approved features should be merged back to the master and the branch deleted.
5. There are config files that you want to change but don’t want to commit, i.e `/app/config/app.php` and `/app/config/database.php` you can prevent yourself from accidentally commiting these files by doing `git update-index --assume-unchanged  /path/to/file.ext` your changes will no longer be tracked. Read more at http://archive.robwilkerson.org/2010/03/02/git-tip-ignorechanges-to-tracked-files/.

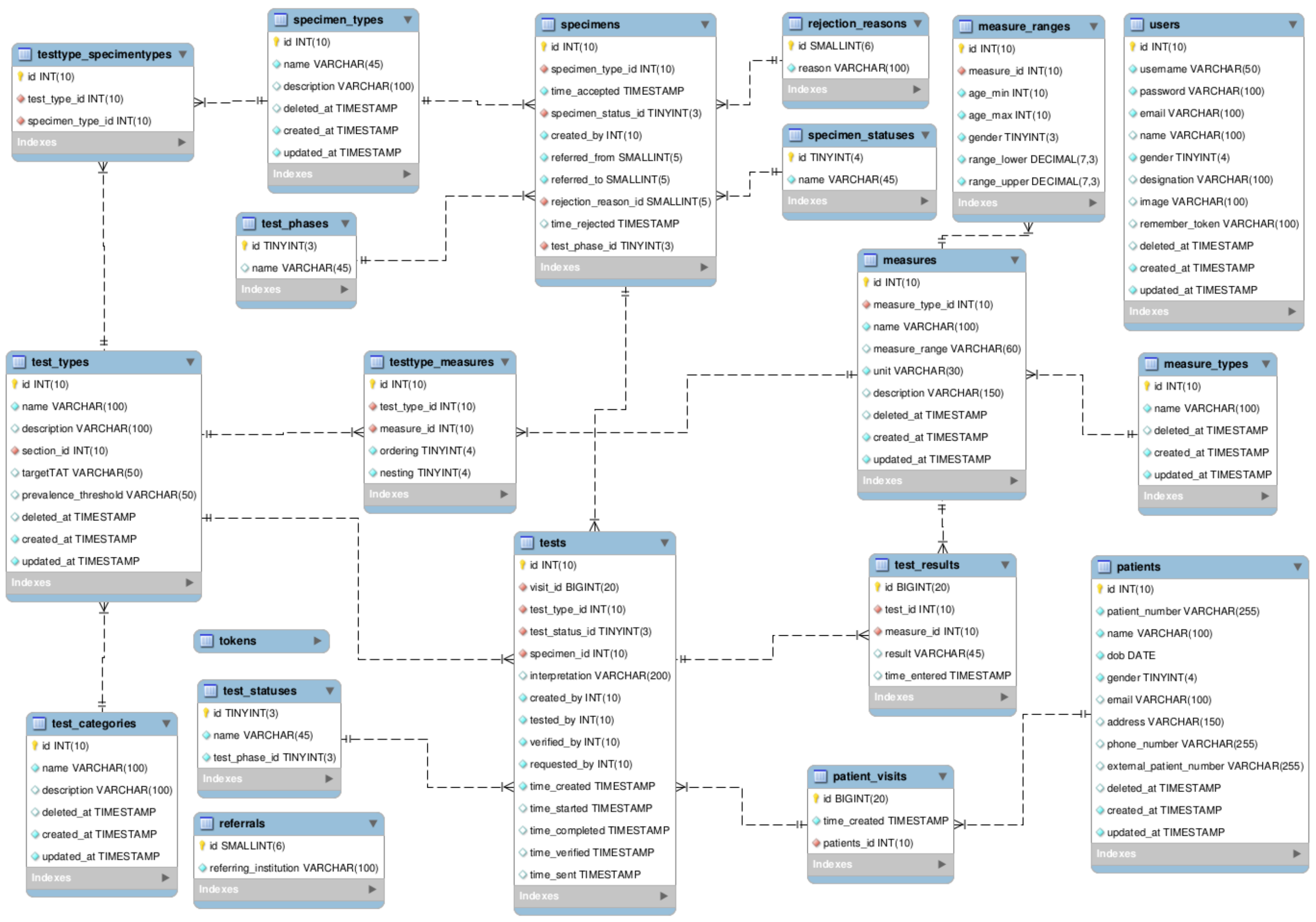
1.5.2 Tests

1. Every feature should have an accompanying test
2. A feature should pass all tests before it is merged into the master branch

1.5.3 Feature check list

1. Tests for the models and controllers where applicable
2. Adding Access controls where applicable
3. Updating the wiki with relevant info about the feature.

Figure 1 Database model



testtype_specimentypes
id INT(10)
test_type_id INT(10)
specimen_type_id INT(10)
Indexes
specimen_types
id INT(10)
name VARCHAR(45)
description VARCHAR(100)
deleted_at TIMESTAMP
created_at TIMESTAMP
updated_at TIMESTAMP
Indexes
specimens
id INT(10)
specimen_type_id INT(10)
time_accepted TIMESTAMP
specimen_status_id TINYINT(3)
created_by INT(10)
referred_from SMALLINT(5)
referred_to SMALLINT(5)
rejection_reason_id SMALLINT(5)
time_rejected TIMESTAMP
test_phase_id TINYINT(3)
Indexes
rejection_reasons
id SMALLINT(6)
reason VARCHAR(100)
Indexes
measure_ranges
id INT(10)
measure_id INT(10)
age_min INT(10)
age_max INT(10)
gender TINYINT(3)
range_lower DECIMAL(7,3)
range_upper DECIMAL(7,3)
Indexes
users
id INT(10)
username VARCHAR(50)
password VARCHAR(100)
email VARCHAR(100)
name VARCHAR(100)
gender TINYINT(4)
designation VARCHAR(100)
image VARCHAR(100)
remember_token VARCHAR(100)
deleted_at TIMESTAMP
created_at TIMESTAMP
updated_at TIMESTAMP
Indexes
specimen_statuses
id TINYINT(4)
name VARCHAR(45)
Indexes
test_phases
id TINYINT(3)
name VARCHAR(45)
Indexes
measures
id INT(10)
measure_type_id INT(10)
name VARCHAR(100)
measure_range VARCHAR(60)
unit VARCHAR(30)
description VARCHAR(150)
deleted_at TIMESTAMP
created_at TIMESTAMP
updated_at TIMESTAMP
Indexes
test_types
id INT(10)
name VARCHAR(100)
description VARCHAR(100)
section_id INT(10)
targetTAT VARCHAR(50)
prevalence_threshold VARCHAR(50)
deleted_at TIMESTAMP
created_at TIMESTAMP
updated_at TIMESTAMP
Indexes
testtype_measures
id INT(10)
test_type_id INT(10)
measure_id INT(10)
ordering TINYINT(4)
nesting TINYINT(4)
Indexes
measure_types
id INT(10)
name VARCHAR(100)
deleted_at TIMESTAMP
created_at TIMESTAMP
updated_at TIMESTAMP
Indexes
tests
id INT(10)
visit_id BIGINT(20)
test_type_id INT(10)
test_status_id TINYINT(3)
specimen_id INT(10)
interpretation VARCHAR(200)
created_by INT(10)
tested_by INT(10)
verified_by INT(10)
requested_by INT(10)
time_created TIMESTAMP
time_started TIMESTAMP
time_completed TIMESTAMP
time_verified TIMESTAMP
time_sent TIMESTAMP
Indexes
test_results
id BIGINT(20)
test_id INT(10)
measure_id INT(10)
result VARCHAR(45)
time_entered TIMESTAMP
Indexes
patients
id INT(10)
patient_number VARCHAR(255)
name VARCHAR(100)
dob DATE
gender TINYINT(4)
email VARCHAR(100)
address VARCHAR(150)
phone_number VARCHAR(255)
external_patient_number VARCHAR(255)
deleted_at TIMESTAMP
created_at TIMESTAMP
updated_at TIMESTAMP
Indexes
tokens
test_statuses
id TINYINT(3)
name VARCHAR(45)
test_phase_id TINYINT(3)
Indexes
test_categories
id INT(10)
name VARCHAR(100)
description VARCHAR(100)
deleted_at TIMESTAMP
created_at TIMESTAMP
updated_at TIMESTAMP
Indexes
referrals
id SMALLINT(6)
referring_institution VARCHAR(100)
Indexes
patient_visits
id BIGINT(20)
time_created TIMESTAMP
patients_id INT(10)
Indexes

INSTRUMENTATION

2.0 Conceptualization

- How do we dynamically add new functionality to handle new analyzers? Using a plugin facility.
- kBLIS can fetch results from different analyzers of the same type? e.g.CELTAC1inWard and CELTAC2 in Ward 2.

Yes! That’s why we have 2 buttons.

is for adding the driver (common to all machines of the same type).

is for specifying the physical location of the machine.

2.1 Implementation

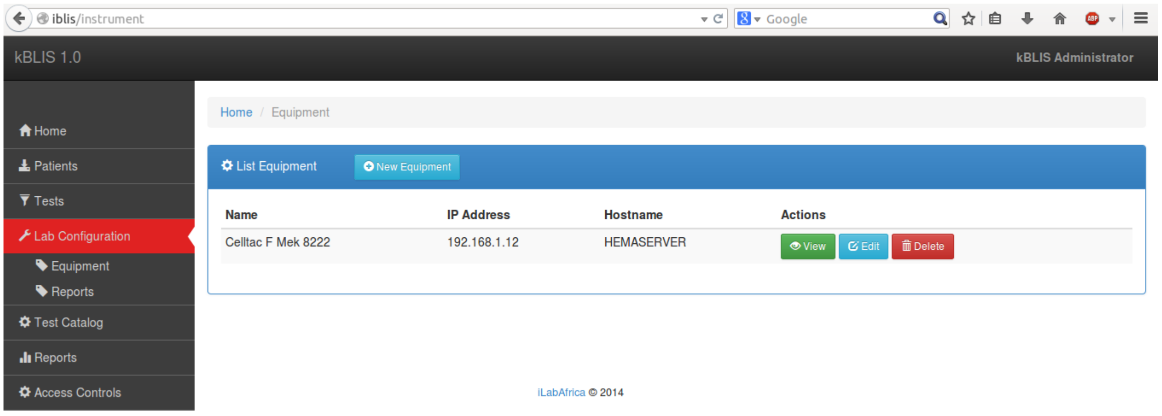
BLIS interfaces with the connected equipment courtesy of a class file that extends the AbstractInstrumentor class which in turn implements the InstrumentorInterface. A sample file is provided (CeltacWBC.php) showing the format in which BLIS expects the result of the getResult() method.On importation, plugin files a copied to the app/kblis/plugins/ directory. From there, they can be loaded into the application via psr-4.

2.2 Instrumentation UI

This can be accessed from the Lab, Configuration >   Equipment sidemenu.

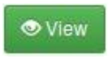
2.3 List Equipment

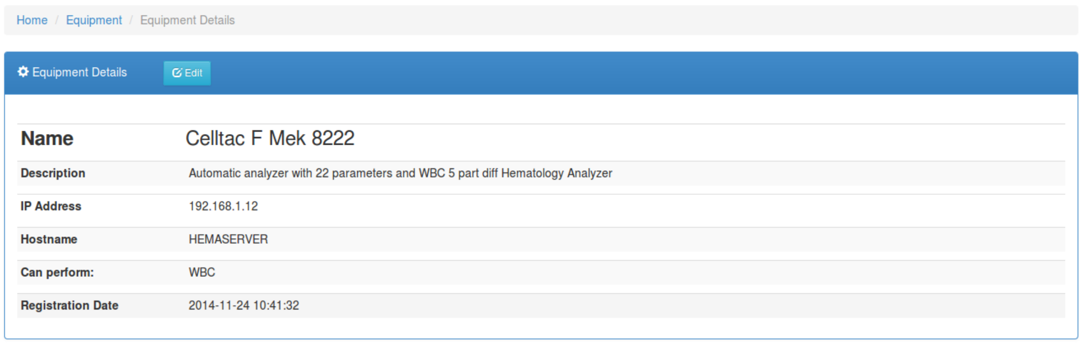
The default page shows a list of all Equipment already configured for use.



Name	IP Address	Hostname	Actions
Celltac F Mek 8222	192.168.1.12	HEMASERVER	View Edit Delete

Figure 2View Equipment Details

- Clicking the  button displays the details of the listed equipment.

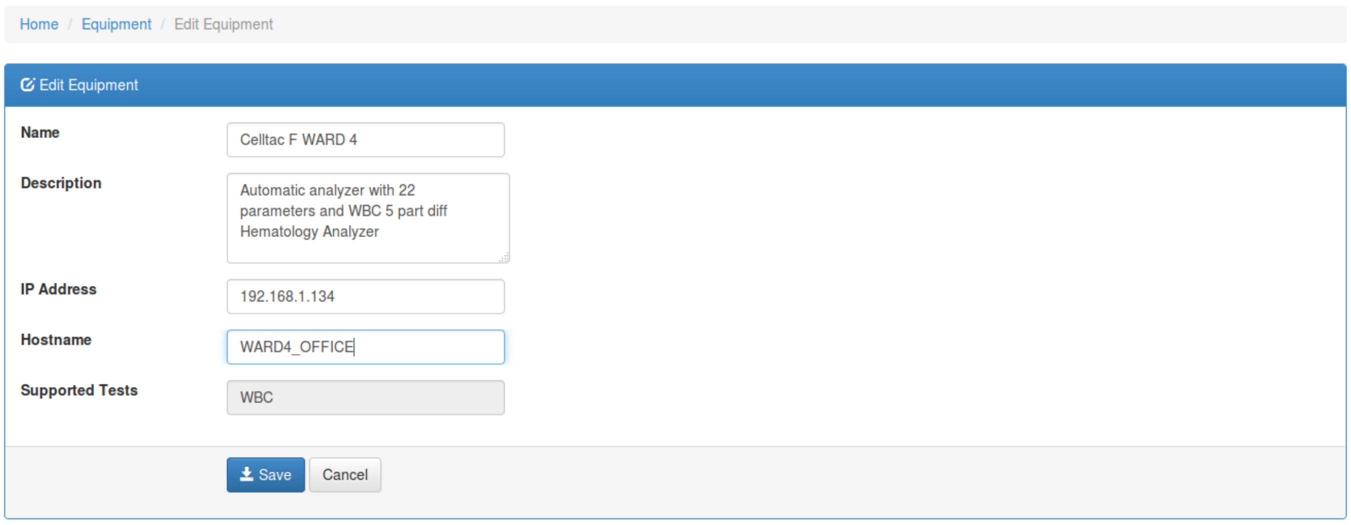


Name	Celltac F Mek 8222
Description	Automatic analyzer with 22 parameters and WBC 5 part diff Hematology Analyzer
IP Address	192.168.1.12
Hostname	HEMASERVER
Can perform:	WBC
Registration Date	2014-11-24 10:41:32

Figure 3Edit Equipment

The button allows for changing the details of the listed equipment including:

- Name
- Description
- Ip address
- Host name


Home / Equipment / Edit Equipment

Edit Equipment

Name	Celltac F WARD 4
Description	Automatic analyzer with 22 parameters and WBC 5 part diff Hematology Analyzer
IP Address	192.168.1.134
Hostname	WARD4_OFFICE
Supported Tests	WBC

Figure 4 Edit Equipment

The Supported Tests field depends on the equipment driver implementation thus is not editable.

2.4 Equipment Drivers

Driver files tell kBLIS how to access testv information from analyzers. A driver implementation should publish following information:The equipment name a unique identification code a description of the equipment.The medical tests it can perform.

2.5 Importing driver

To import a driver file,

1. Click on the New Driver button on the Equipment List page.
2. In the dialog box that appears, click the **Browse** button to open a files dialog box.

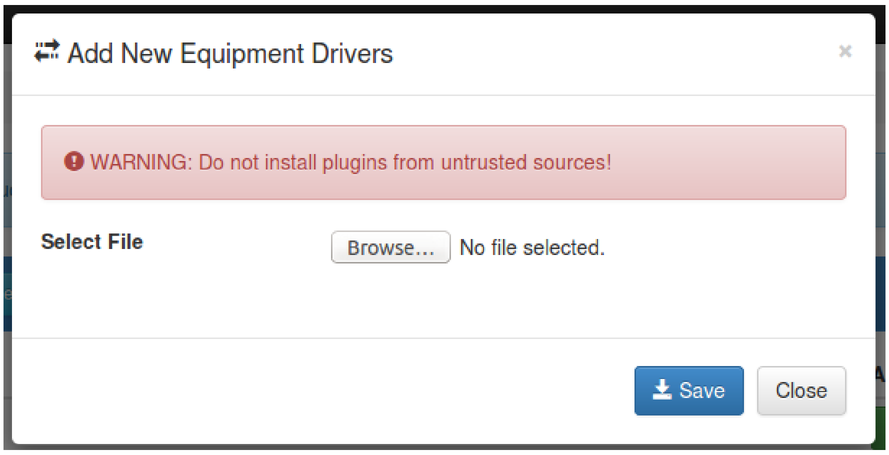

Add New Equipment Drivers
WARNING: Do not install plugins from untrusted sources!
Select File
Browse…
No file selected.
Save
Close

Figure 5Adding new driver

3. Select a driver file

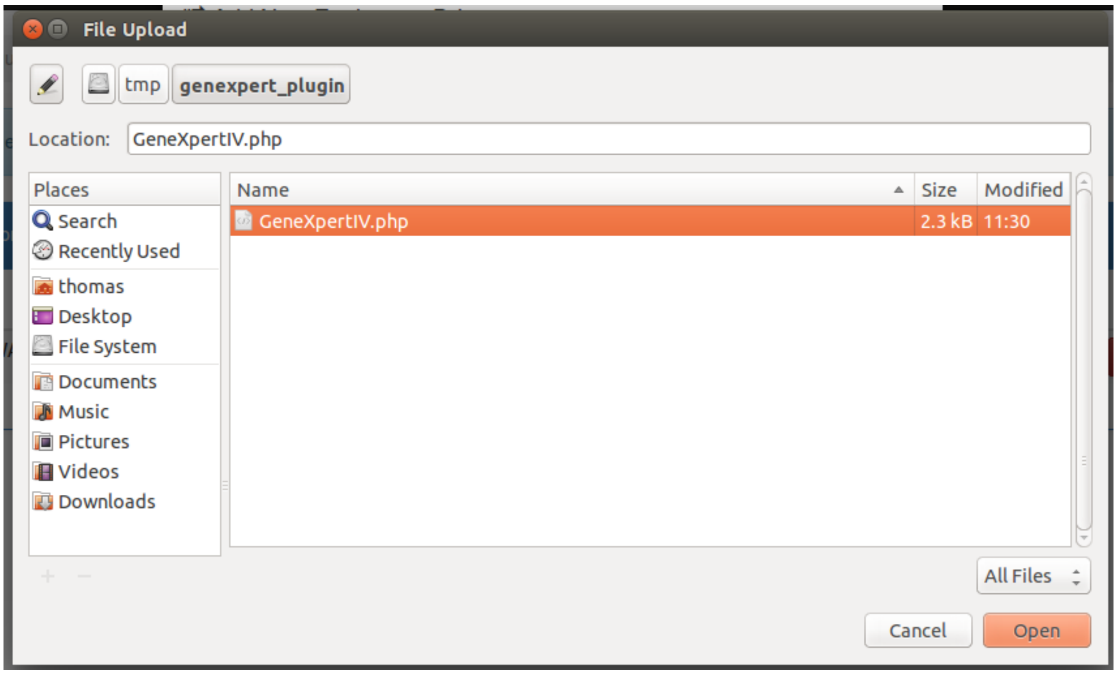
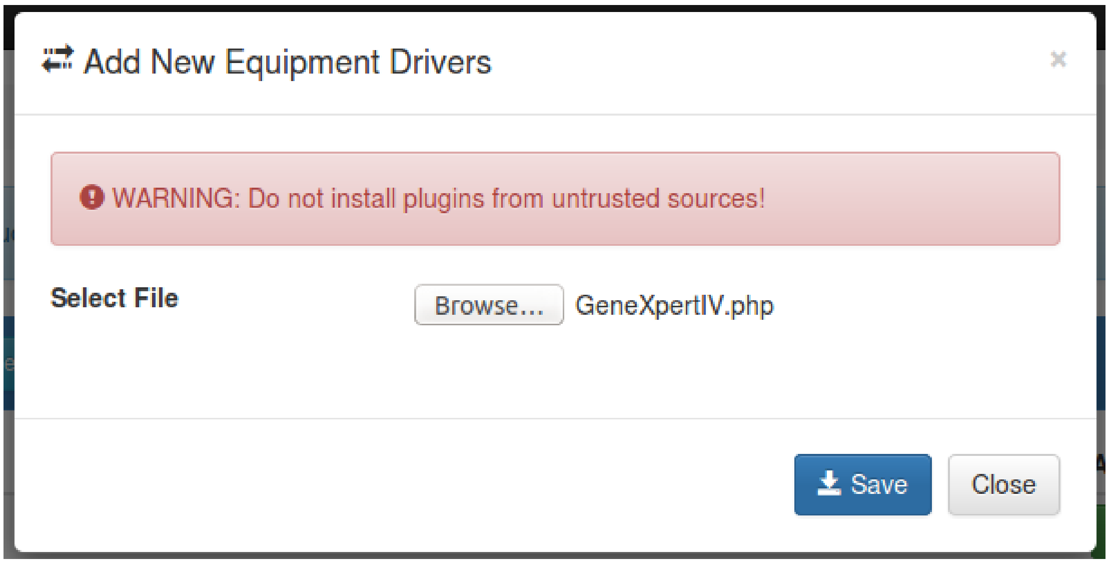


Figure 6 Select Driver

4. Save the driver file

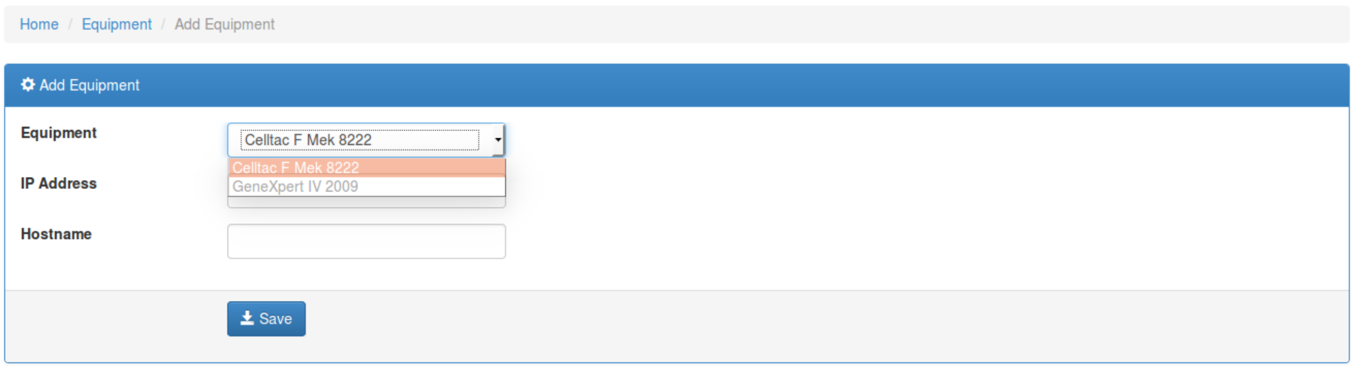

Add New Equipment Drivers

WARNING: Do not install plugins from untrusted sources!

Select File — Browse… GeneXpertIV.php

Figure 7 saving a new driver

The new driver will now be listed as one of the available drivers.

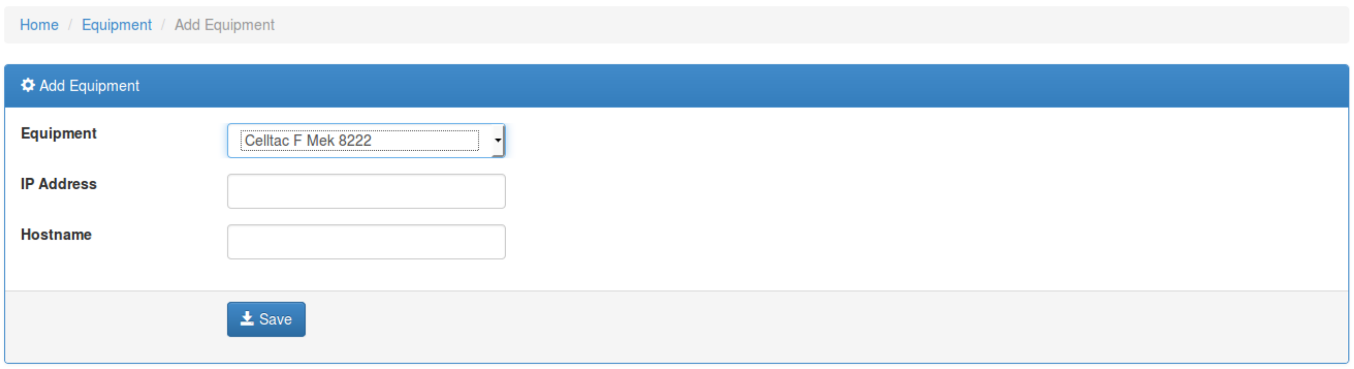


Equipment	Celltac F Mek 8222
	Celltac F Mek 8222
	GeneXpert IV 2009
IP Address	
Hostname	

Figure 8 new driver added to list of drivers

2.6 Add New Equipment

In this step, the user defines the location of a particular machine by specifying the IP address and optionally the hostname to which the machine is attached. The process is initiated by clicking on New Equipment from the Equipment List page.



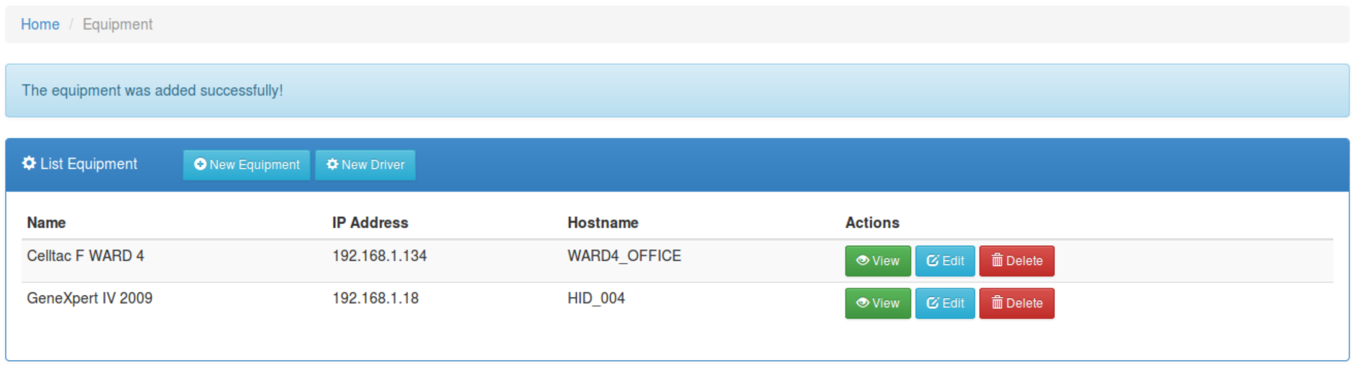
Home / Equipment / Add Equipment

Add Equipment

Equipment	Celltac F Mek 8222
IP Address	
Hostname	

Figure 9 Add new equipment

On saving, the new equipment is now listed as one of those available on the Equipment List page.



Home / Equipment

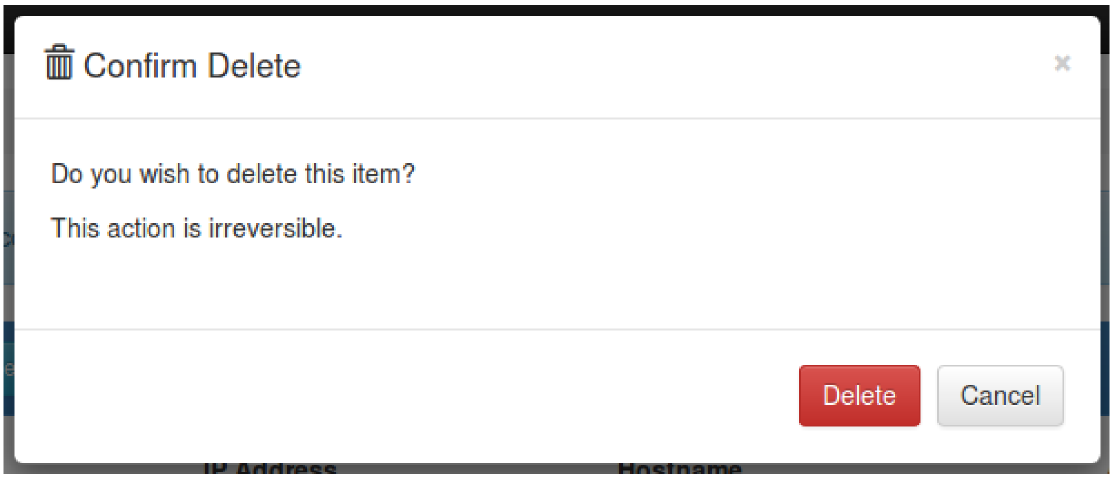
The equipment was added successfully!

Name	IP Address	Hostname	Actions
Celltac F WARD 4	192.168.1.134	WARD4_OFFICE	View Edit Delete
GeneXpert IV 2009	192.168.1.18	HID_004	View Edit Delete

Figure 10 list of equipment

2.7 Delete existing equipment

- Click the Delete button on the equipment index page
- Click **Delete** in the resultant dialog box



Confirm Delete

Do you wish to delete this item?

This action is irreversible.

INTERFACING

This module isused to interface between different systems and BLIS. Currently a Sanitas implementation exists.

3.0 How to use

Due to the reliance of an implementation, the specific interfaces must have methods retrieve() , process() and send()

3.0.1 retrieve ()

- To in the case of Sanitas retrieving works behind the scenes, data will be retrieved automatically.
- For medboss, the interfacer must listen for when a user searches for a specific number or name, a call to interfacer::retrieve() will then handle going to the mssql db and retrieving the specific details.

3.0.2 send ()

In both case sending the result is as simple as calling Interfacer::send(testId).

For more technical details see#79 #87 #94 for more details, and check in the app/api/ folder.

INVENTORY

4.0 Introduction

The inventory module deal with the laboratory commodities from when they are received from suppliers to issuing to different locations within the hospital as well as stock taking (capturing the physical count) of the laboratory commodities. In order to have a standard naming for commodities, units of issue and suppliers, there are three interfaces which include commodities, metrics and suppliers respectively where the name and descriptions are captured.

Commodities are the drugs, reagents and equipment's required in the laboratory. Metrics or unit of issue is the unit or quantity in which commodities or items are procured, stored, and issued. For example in kilograms, milligrams (100mg) etc. Suppliers are the companies or organizations that supply the laboratory with commodities.

4.1 Components

The module consists of three components Laboratory Stock Card, Laboratory Top-Up Form and Laboratory Stock Take.

4.2 Laboratory Stock Card

This component deals with receipts and issues. Receipts are the laboratory commodities that the hospital receives from suppliers. Issues are the laboratory commodities that are given out to various points within the hospital. A user cannot issue a commodity that has not yet been received or whose quantity is 0. The quantity received and quantity issued is critical in determining the stock balance.

4.3 Laboratory Top-Up Form

This component deals with making request for more commodities to be supplied to the laboratory. It consists of the commodities, how much of each is available, how much had been issued and how much is required (order quantity).

4.4 Laboratory Stock Take

This component deals with taking a physical count of the commodities available at the laboratory and comparing it with the quantity remaining (stock balance) in the system. Stock taking can be done on a monthly and quarterly basis. Any discrepancies are captured as well as the total price of the commodities physically counted.

LOCALIZATION

5.0 Introduction

Localization is localization. : point_left :

Basically all words should be in the app/lang folder. This enables easier translations to other languages aka localization.

All words in the Views should be placed in *messages.php*, custom validation messages in *validation.php*

5.1 How to use

1. The locale can be changed programatically or in the file /app/config/app.php from the default (en) to any other available in the /app/lang folder.
2. Open the messages.php and places your word 'login' => 'Login',

• In the view do {{trans('messages.login')}} which will output Login

1. Pluralization? In messages.php add 'user' => 'User|Users'

• In the view do {{ Lang::choice('messages.permission', 2) }} the second parameter is if you want plural(2+) or single(1)

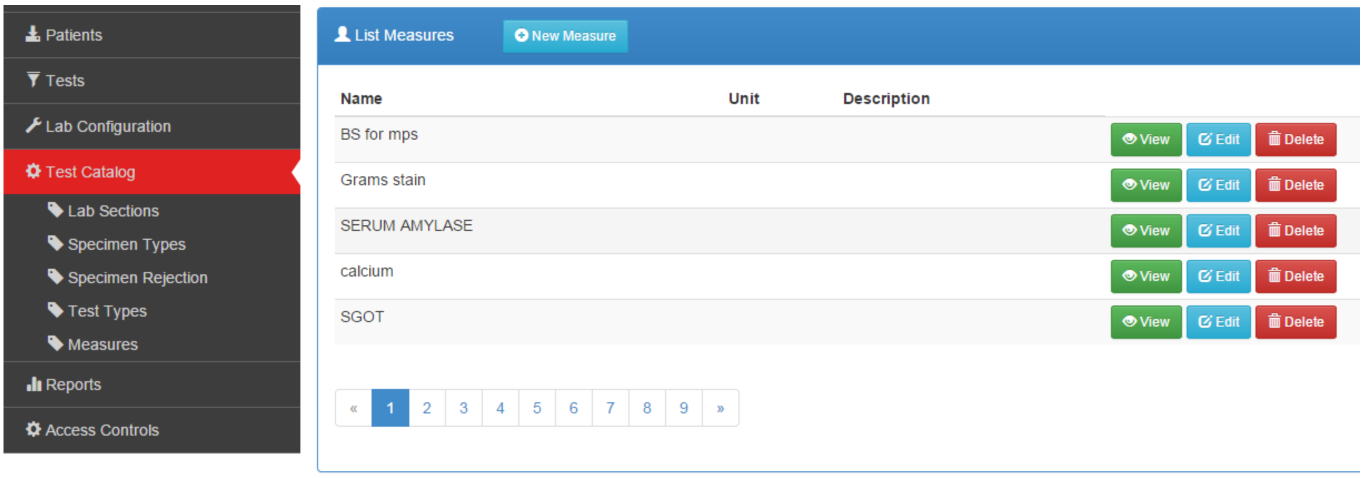
MEASURES

6.0 DB Structure

Latest Change: The measure ranges (previously on the measure table) and interpretations are in the measure_ranges table

6.1 Measure UI

Allows you to add and edit measures, and specify the interpretation for multiple ranges of measure values for each measure.



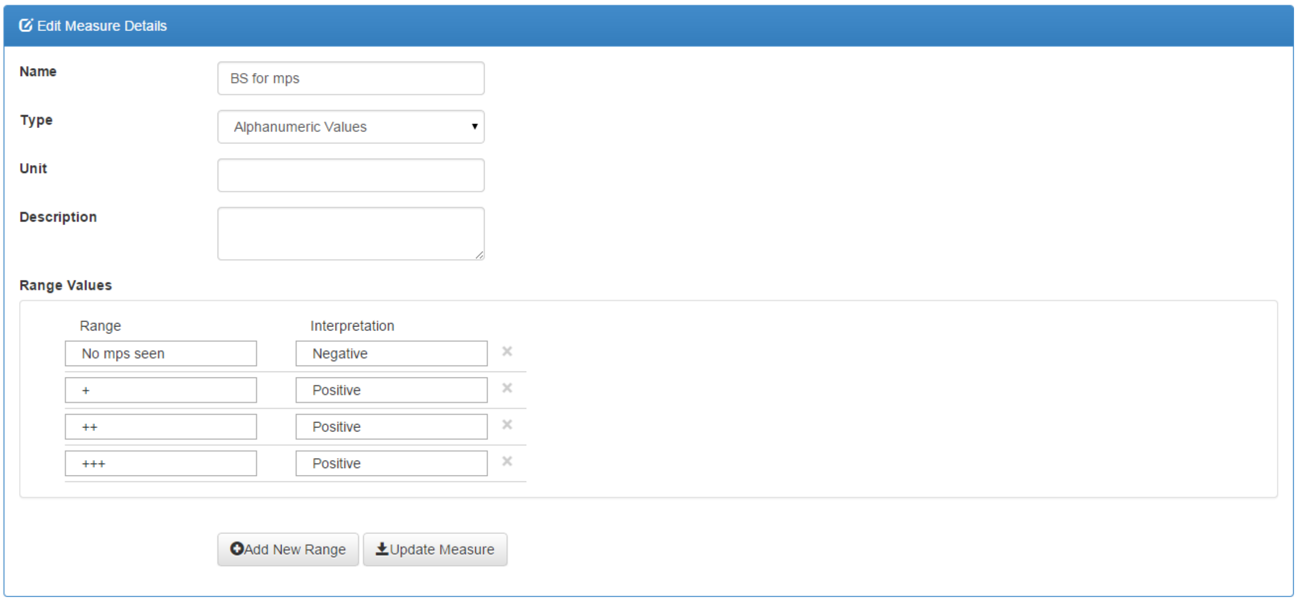
Name	Unit	Description
BS for mps		
Grams stain		
SERUM AMYLASE		
calcium		
SGOT		

Figure 11 Measure UI

6.1.1 Adding a New Measure

To add click the **New Measure** button to see:

For Alphanumeric ranges


Edit Measure Details

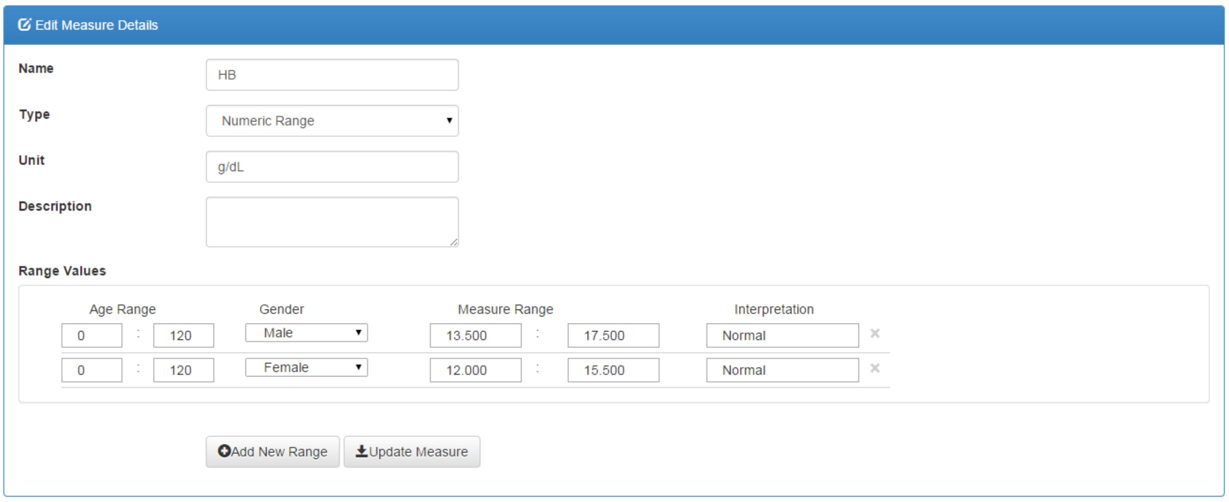
Name	BS for mps
Type	Alphanumeric Values
Unit	
Description	

Range Values

Range	Interpretation
No mps seen	Negative
+	Positive
++	Positive
+++	Positive

Figure 12 Adding new Alphanumeric measure

For Numeric Ranges


Edit Measure Details

Name	HB
Type	Numeric Range
Unit	g/dL
Description	

Range Values

Age Range		Gender	Measure Range		Interpretation
0	120	Male	13.500	17.500	Normal
0	120	Female	12.000	15.500	Normal

Figure 13 add new numeric range

Add the measure name, type, unit (for numeric ranges), description (optional) and range values with their interpretations (where applicable) Click Add New Range to add a measure range and Update Measure to save changes

6.1.2 Editing a Measure

To edit click the Edit button of an entry change the measure name, type, units, description and range values with their interpretations (where applicable) as required.

6.1.3 Deleting a Measure

To delete click the Delete button of an entry

MEDICAL TESTS

7.0 Introduction

Different medical systems have distinct stages and processes of conducting medical test. This document will track the life cycle of a medical test on-board iBLIS. After receiving test request from a patient or other embedded systems the immediate step is to book for the test via the laboratory system.

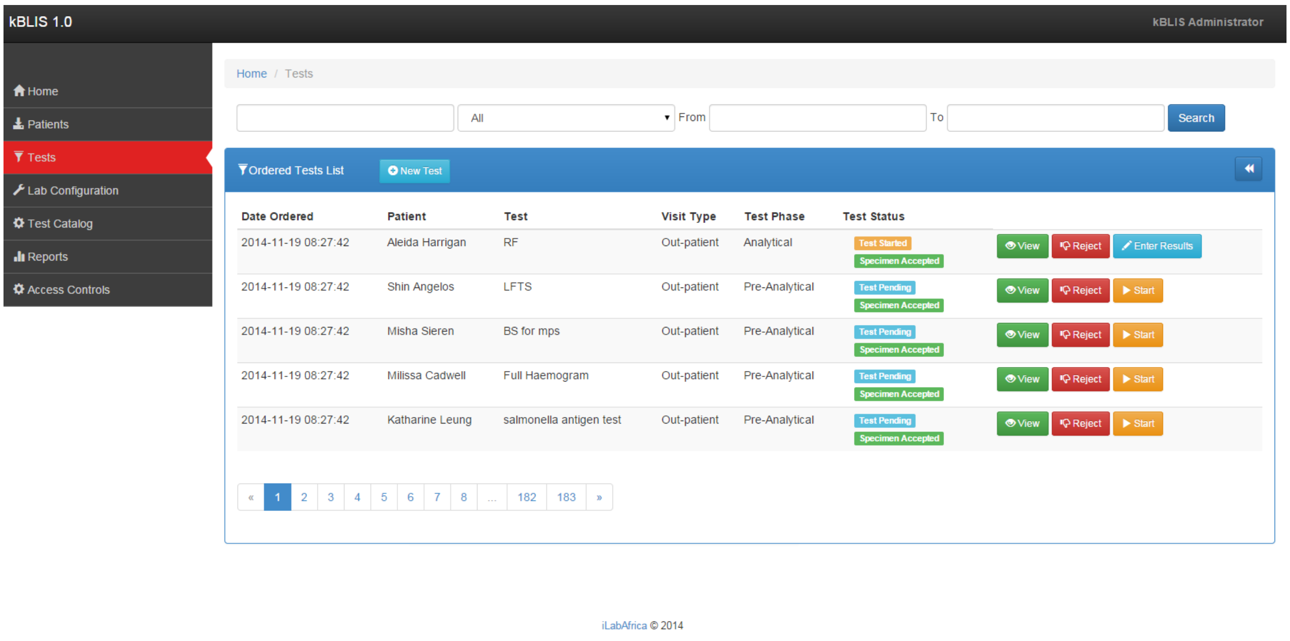
The initial stage of capturing the test details is most important stage. In essence, accurate data capturing will streamline the logical steps to ensure that what you test is what you meant to test, and that the final test report meets the client’s needs and expectations.

7.1 Ordering for a Test

For a test order to be initiated the patient must be registered in the system via the patient interface. This will provide the patient details which in-turn will be used for ordering the test. Test order can also be requested from an external system.

7.2 Test Booking UI

Below is the first interface during thetesting process. There several search fields on UI. The first one allow you to filter the test based on the test type, the second one will allow you to filter using test status and third one will use date as the filtering criteria. It includes the already ordered tests, tests status and a New Test button for initializing new test order


kBLIS 1.0 kBLIS Administrator

Home / Tests

Ordered Tests List — New Test

Date Ordered	Patient	Test	Visit Type	Test Phase	Test Status
2014-11-19 08:27:42	Aleida Harrigan	RF	Out-patient	Analytical	Test Started, Specimen Accepted
2014-11-19 08:27:42	Shin Angelos	LFTS	Out-patient	Pre-Analytical	Test Pending, Specimen Accepted
2014-11-19 08:27:42	Misha Sieren	BS for mps	Out-patient	Pre-Analytical	Test Pending, Specimen Accepted
2014-11-19 08:27:42	Milissa Cadwell	Full Haemogram	Out-patient	Pre-Analytical	Test Pending, Specimen Accepted
2014-11-19 08:27:42	Katharine Leung	salmonella antigen test	Out-patient	Pre-Analytical	Test Pending, Specimen Accepted

Figure 14 Test Booking User Interface

7.2.1 New Test order

The iBLIS system can accept and initiate test requests through different approaches.

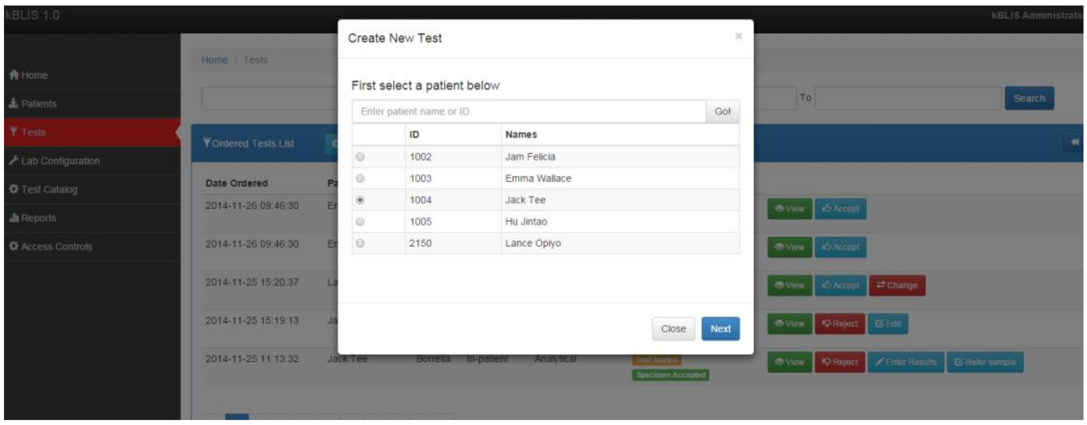
i) By use of the Tests Link

The step below shows how ordering a test from the default tests page can be achieved.

1. Click on the Tests link on the side bar navigation menu on the left of the screen.

2. Click the New Test button to launch the test order.

3. Select the desired patient by tying the patient Id or Name or by clicking the GO button to list the available patients.

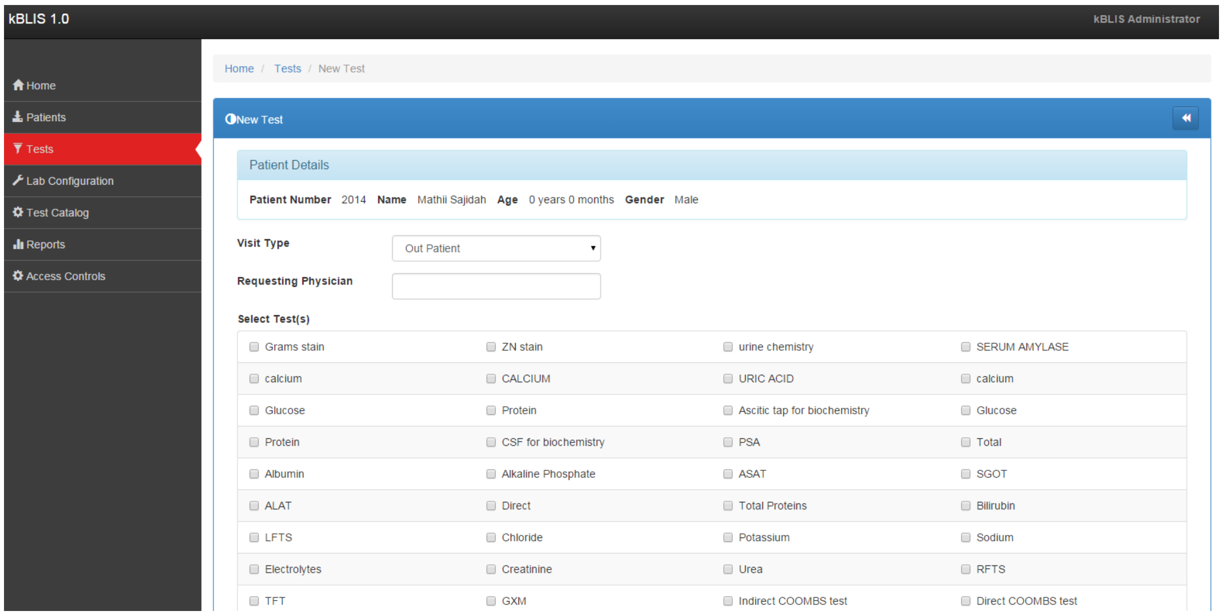

Create New Test

First select a patient below

	ID	Names
○	1002	Jam Felicia
○	1003	Emma Wallace
●	1004	Jack Tee
○	1005	Hu Jintao
○	2150	Lance Opiyo

Figure 15 patient test list

4. In the New test page, input the visit type ,the requesting physician and from the listed tests select the desired test(s).



Patient Details

Patient Number 2014 **Name** Mathii Sajidah **Age** 0 years 0 months **Gender** Male

Visit Type Out Patient

Requesting Physician

Select Test(s)

☐ Grams stain	☐ ZN stain	☐ urine chemistry	☐ SERUM AMYLASE
☐ calcium	☐ CALCIUM	☐ URIC ACID	☐ calcium
☐ Glucose	☐ Protein	☐ Ascitic tap for biochemistry	☐ Glucose
☐ Protein	☐ CSF for biochemistry	☐ PSA	☐ Total
☐ Albumin	☐ Alkaline Phosphate	☐ ASAT	☐ SGOT
☐ ALAT	☐ Direct	☐ Total Proteins	☐ Bilirubin
☐ LFTS	☐ Chloride	☐ Potassium	☐ Sodium
☐ Electrolytes	☐ Creatinine	☐ Urea	☐ RFTS
☐ TFT	☐ GXM	☐ Indirect COOMBS test	☐ Direct COOMBS test

Figure 16 Test list

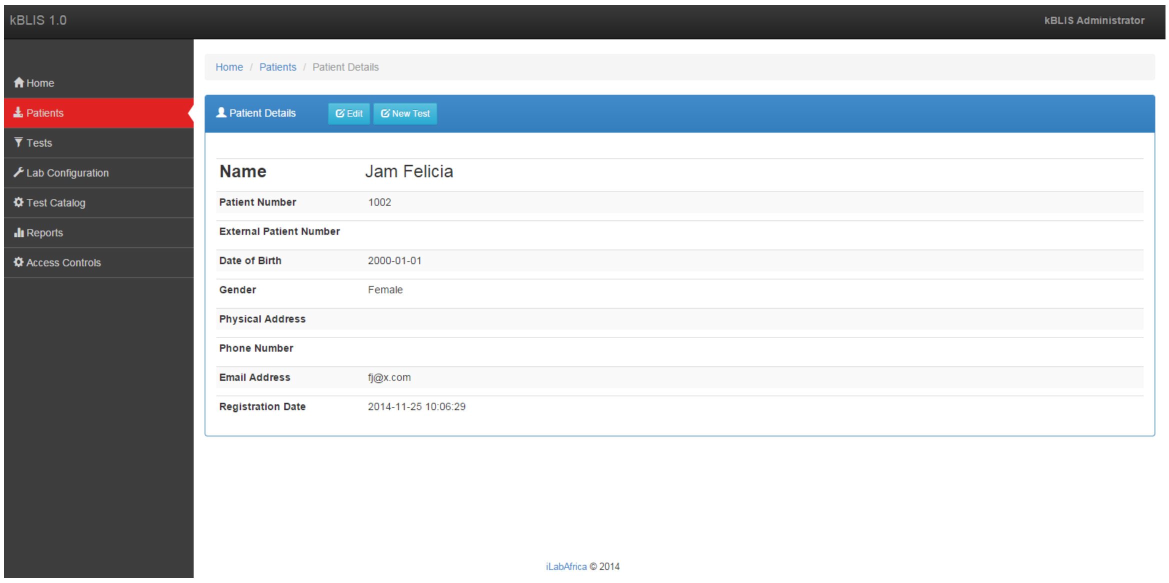
5. Click the save button.

ii). By use of the patient's details page

A test can be booked directly from the patient details page. This page can be accessed after selecting the patient at subject of the tests.

1. Click on the Patients link on the side bar navigation menu on the left of the screen.

2. Identify the desired patient and click the corresponding View button to load the Patient Details page.


kBLIS 1.0

kBLIS Administrator

Name	Jam Felicia
Patient Number	1002
External Patient Number	
Date of Birth	2000-01-01
Gender	Female
Physical Address	
Phone Number	
Email Address	fj@x.com
Registration Date	2014-11-25 10:06:29

iLabAfrica © 2014

Figure 17Example of patient details

3. On the Patient Details page, click the New Test button.

iii). By use of an external system

A test can also be booked via an external system. iBLIS will have a module through which a test order can be invoked by multiple external systems. This will only happen if the systems can integrate and map fully with iBLIS. External system configuration and integration with iBLIS will depend on the available systems in any given hospital. All the embedded systems will fully be tested to ensure a smooth running of iBLIS and the particular system at stake before inter-dependability can be authorized.

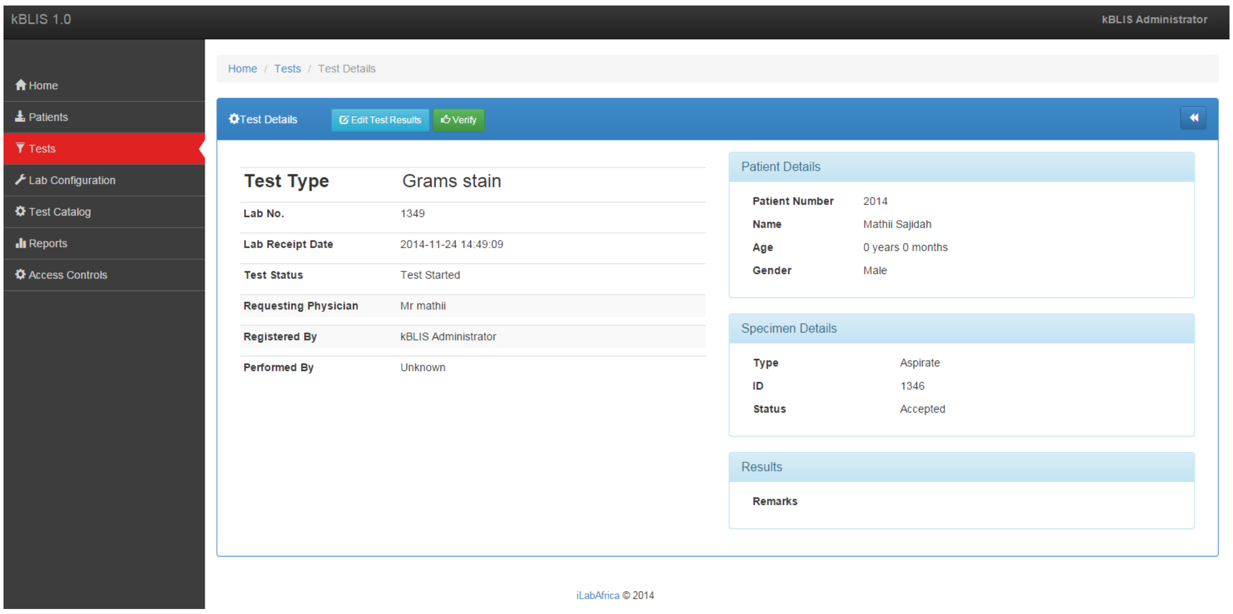
7.2.2 Test Details

This UI is used for viewing the details of a test and making any changes which might have been captured wrongly during the test process.It is also shows the all the changes made right from the time a new test is ordered till the test results are verified. These processes accompanying these changes include; Test Status Accept or reject test, accept or reject a specimen, start a test, entering results, rejecting results and test verification.

1. Click on the Tests link on the side bar navigation menu on the left of the screen.

2. Select the test you want to view.

3. Click on the View button to proceed to test details page.


kBLIS 1.0 | kBLIS Administrator

Test Type	Grams stain
Lab No.	1349
Lab Receipt Date	2014-11-24 14:49:09
Test Status	Test Started
Requesting Physician	Mr mathii
Registered By	kBLIS Administrator
Performed By	Unknown

Patient Details

Patient Number	2014
Name	Mathii Sajidah
Age	0 years 0 months
Gender	Male

Specimen Details

Type	Aspirate
ID	1346
Status	Accepted

Results

Remarks

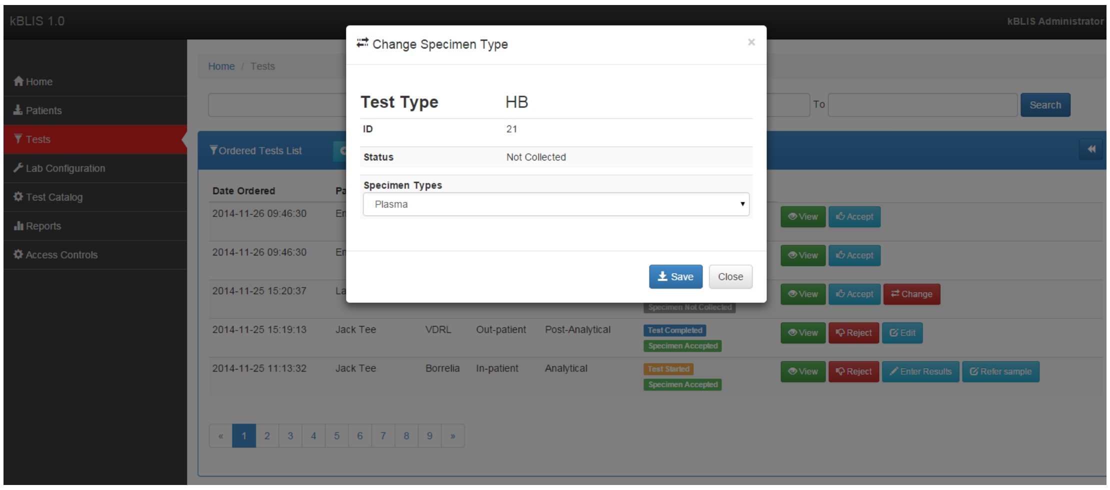
iLabAfrica © 2014

Figure 18 Test details

7.2.3 Specimen Change

Once a test has been created the next stage is to collect specimen. The specimen is collected and verified. The type of the specimen sample can vary from one test to another. To change from one specimen type to another follow the steps below;

1. Click on the Tests link on the side bar navigation menu on the left of the screen.
2. Select the test of interest and check where Test Status is specimen not collected.
3. Click on the Change button and choose the desired specimen type.



Change Specimen Type

Test Type	HB
ID	21
Status	Not Collected
Specimen Types	Plasma

Save Close

Date Ordered					
2014-11-26 09:46:30					
2014-11-26 09:46:30					
2014-11-25 15:20:37					Specimen Not Collected
2014-11-25 15:19:13	Jack Tee	VDRL	Out-patient	Post-Analytical	Test Completed Specimen Accepted
2014-11-25 11:13:32	Jack Tee	Borrelia	In-patient	Analytical	Test Started Specimen Accepted

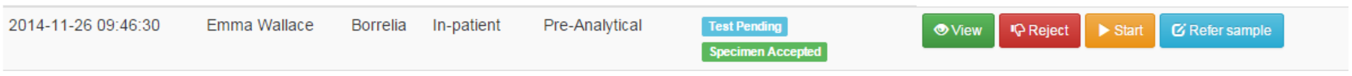
Figure 19 Change specimen

4. Click Save.

7.2.4 Specimen Acceptance

Before the testing process is started the acceptance of the specimen must be registered in the IBLIS system. This will be achieved through;

1. Click on the Tests link on the side bar navigation menu on the left of the screen.
2. Select the test of interest and check where Test Status is specimen not collected.
3. Click on the Accept button to acknowledge the specimen.
4. This will change the specimen status to *started* and the button to Start the test will show.

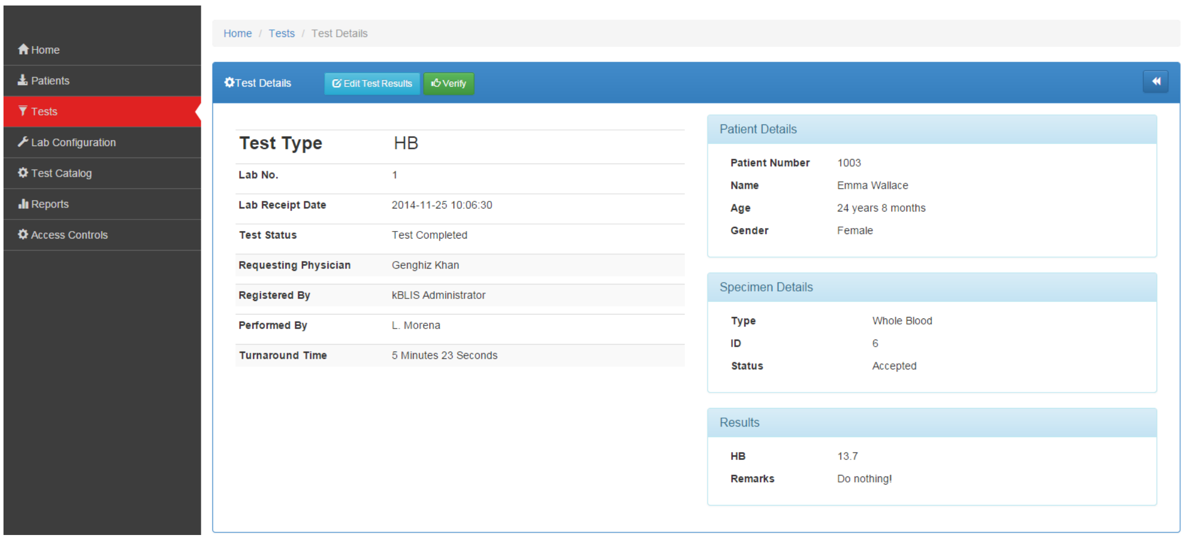


2014-11-26 09:46:30	Emma Wallace	Borrelia	In-patient	Pre-Analytical	Test Pending Specimen Accepted

7.2.5 Specimen Rejection

Specimen can be rejected on several levels in iBLIS. The first level is during specimen collection. Mistakes can be made hence making the specimen invalid. On other occasions the specimen samples can be damaged or contaminated during testing. This too invalidates the specimen.Finally,the specimen can be rejected because the test results do not conform with what is expected. If the collected specimen cannot progress to testing and actualize the test due to various reasons it can then be rejected through;

1. Click on the Tests link on the side bar navigation menu on the left of the screen.
2. Select the test of interest and check where Test Status is specimen accepted.
3. Click on the Reject button to disapprove the specimen.



Test Type	HB
Lab No.	1
Lab Receipt Date	2014-11-25 10:06:30
Test Status	Test Completed
Requesting Physician	Genghiz Khan
Registered By	kBLIS Administrator
Performed By	L. Morena
Turnaround Time	5 Minutes 23 Seconds

Patient Details

Patient Number	1003
Name	Emma Wallace
Age	24 years 8 months
Gender	Female

Specimen Details

Type	Whole Blood
ID	6
Status	Accepted

Results

HB	13.7
Remarks	Do nothing!

Figure 20 Test type

7.2.6 Starting a Test

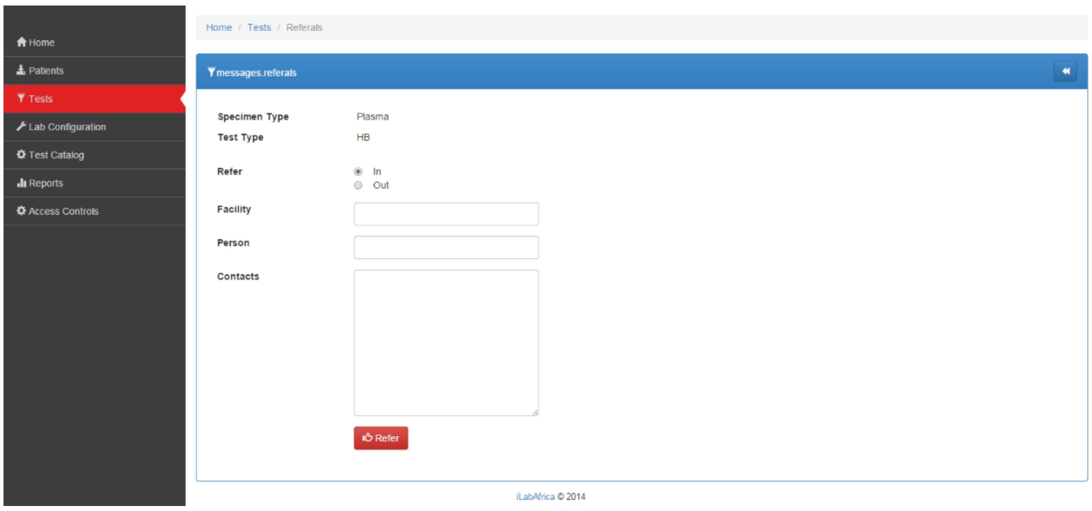
A test can only be started if the collected specimen is authentic. Once the specimen is verified and allowed to proceed to testing. The testing process can be given a green light to commence through;

1. Click on the Tests link on the side bar navigation menu on the left of the screen.
2. Select the test of interest and check where Test Status is Specimen Accepted.
3. Click on the Start button to commence the testing process.

7.2.7 Referring a Test

In some cases the hospital may not have the capability to actualize some of tests. This will result to the test being referred to another medical institution for testing. The accepted specimen can be referred via;

1. Click on the Tests link on the side bar navigation menu on the left of the screen.
2. Select the test of interest and check where Test Status is Test Pending and specimen is accepted.
3. Click on the Refer sample button.



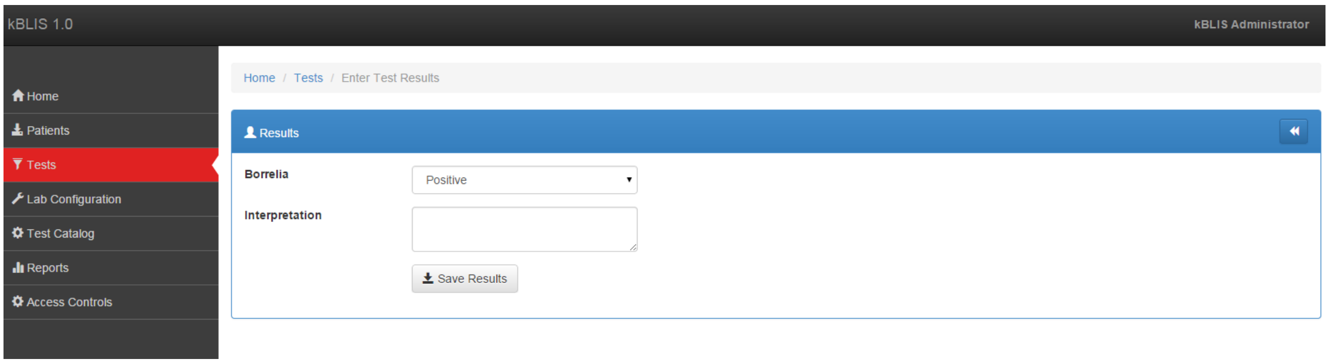
Specimen Type	Plasma
Test Type	HB
Refer	[✓ In] Out
Facility	
Person	
Contacts	

Figure 21 Refer sample

7.2.8 Test Results Acceptance and Entering

After a successful test, the results are then recorded into iBLIS. The page can be accessed via;

1. Click on the Tests link on the side bar navigation menu on the left of the screen.
2. Select the test that whose results are inn.
3. Click on the Enter Results button to feed the results into the system.



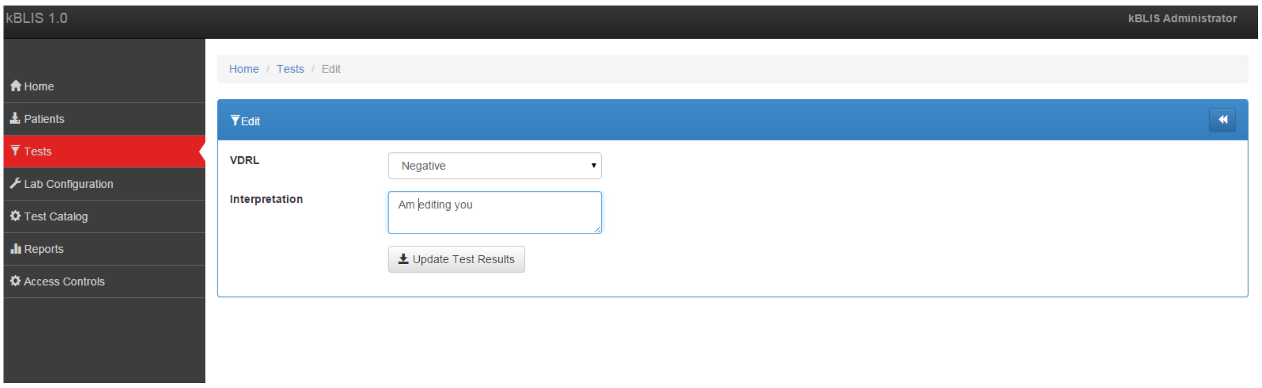
Borrelia: Positive

Interpretation:

7.2.9Editing Test Results

If the test results were entered incorrectly they can be edited. This can only be done before test verification. Follow the steps below to edit test results;

1. Click on the Tests link on the side bar navigation menu on the left of the screen.
2. Select the test whose results are to be edited and test status is completed.
3. Click on the Edit button to make the changes.



VDRL: Negative

Interpretation: Am editing you

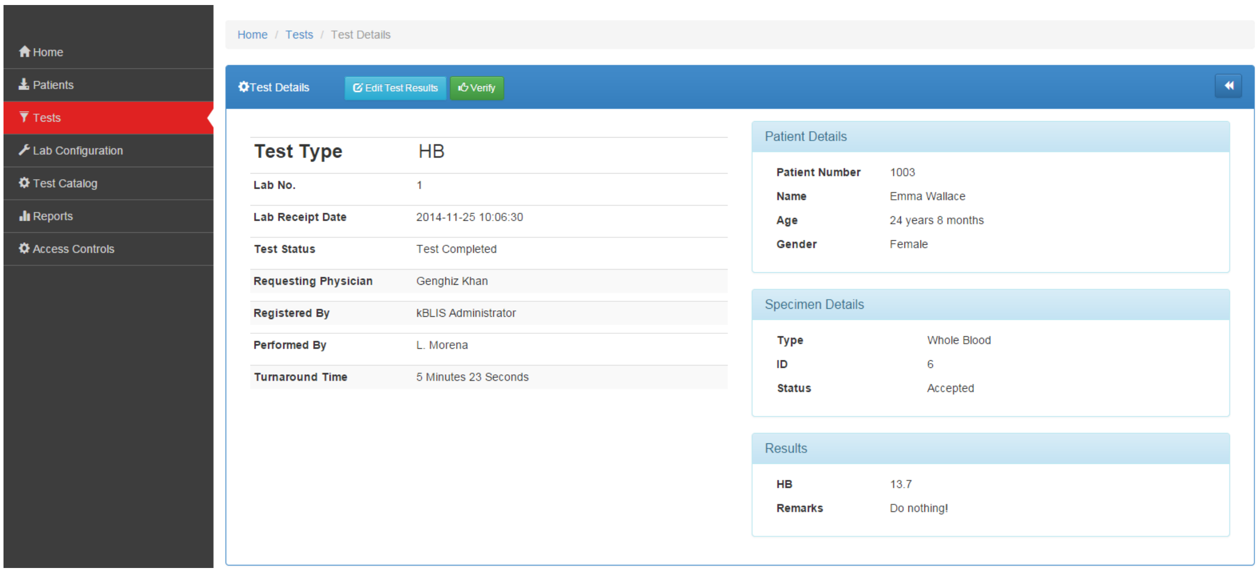
Figure 22 Edit test

7.2.9.1Test Verification

A test can only be verified if all the above processes and conditions are correctly actualized. The verification will be done by an experienced medical personnel. The results will be compared against preset criteria designed for verifying the particular test.

Once the test is verified, is can be given verified indicator on the system by;

1. Click on the Tests link on the side bar navigation menu on the left of the screen.
2. Select the test whose results are to be verified and test status is completed.
3. Click on the Verify button to proceed to the test details page.
4. Click on the Verify button on the test details page to commit the verification.



Test Type	HB
Lab No.	1
Lab Receipt Date	2014-11-25 10:06:30
Test Status	Test Completed
Requesting Physician	Genghiz Khan
Registered By	kBLIS Administrator
Performed By	L. Morena
Turnaround Time	5 Minutes 23 Seconds

Patient Details

Patient Number	1003
Name	Emma Wallace
Age	24 years 8 months
Gender	Female

Specimen Details

Type	Whole Blood
ID	6
Status	Accepted

Results

HB	13.7
Remarks	Do nothing!

Figure 23 Test Verification

PERMISSIONS AND ROLES

8.0 Introduction

Permissions and roles prevent certain users from doing certain things. Entrust is a package that helps with this.

8.1 How to use

- Check if user has a role? $user->hasRole('Technician')
- Check if user has a permission? $user->can('verify_test')
- Check if user has a role and permission? $user->ability(['Doctor'], ['can_edit'])
- Limit routes to a specific permission, there is a new checkPerms filter, pass it the permission, as shown below

array("before" => "checkPerms:manage_users") ...

QUALITY CONTROLS

9.0 Introduction

Quality controls provide a way to assure the lab technologists that the machines are running as they should. It involves testing the machines to see if it produces consistent results day in day out. This is achieved by running tests with sample which the results are already known beforehand.

9.1 How the module works

There are four components to this module

1. Lots
2. Controls
3. Results entry
4. Reporting

9.1.1 Lots

A Lot is a large quantity of reagent that can be used to run the controls over a long period. Each lot may have different ranges, thus the need to batch them into lots.

9.1.2 Controls

A control is the sample to be tested. Controls have a:-

- A name
- Instrument they are connected to
- Measures with ranges (Numeric or alphanumeric)
- Expected values
- Units

These are used to produce the reports and the Levey Jennings chart (Coming soon).

9.1.3 Results entry

Just entering the results for the controls defined above.

9.1.4 Reporting

The tabular report lists all the tests done on a specific control for specified dates.

NB: Since the measure range change over time, the only thing the user needs to do is update the ranges which will create new entries in the ranges table thus preserving the historical data of previously entered results.

TESTING

10.0 Introduction

Testing is a way of ensuring your code works and that when it breaks you will know. Ideally you should test all your methods, code coverage should be 100 % percentege.

The tests have been hooked up to Travis which is a service that pulls our repo, runs the commands on travis.yaml and runs the test. If the tests don't pass an email will be sent to whoever made the commit (configurable).

10.1 DB

The DB is in memory sqlite, its harder, better, faster stronger. Access just as you would mysql as eloquent abstracts the dirty details.

10.2 How to Test

Currently we are testing controllers, since most of the functionality lives there.

• To do: Test models and views

Here are a few steps to follow when testing, this is not exhaustive please experiment with other ways.

1. Create your test class and extend Testcase.
2. Initialize you variable and other stuff in setup( ) i.e

Artisan::call('migrate') and Artisan::call('db:seed')

1. For sending POST to a controller method do

$this->action('POST', 'HomeController@index', array('name'=>'Vito Corleone')) 1. Does the controller method require a parameter, no worries just put the named parameter to your array, laravel magically handles that.

$this->action('POST', 'HomeController@index', array('name'=>'Vito Corleone', 'id'=>'235'))

1. Some of the things to test for

- assertRedirectedToRoute() test that your function redirected properly
- assertSessionHasErrors('email') check that the session has errors for a particular key.

Generally create the test then make the functionality, if you get an error create a test that fails, fix the error and make sure the test passes.