STEM App Documentation

Specifaction of extended functionality, CONCEPTS & issues, As of 30/08/2019

Matthew Long

STEM app

Contents

[Concept/design 2](#_Toc17989691)

[Basic diagram 2](#_Toc17989692)

[Platforms and tools 3](#_Toc17989693)

[List of extra features extended from the base ‘Google Science Journal’ app: 4](#_Toc17989694)

[Connection setup windows. 4](#_Toc17989695)

[Window one 4](#_Toc17989696)

[Window two 4](#_Toc17989697)

[Window three 4](#_Toc17989698)

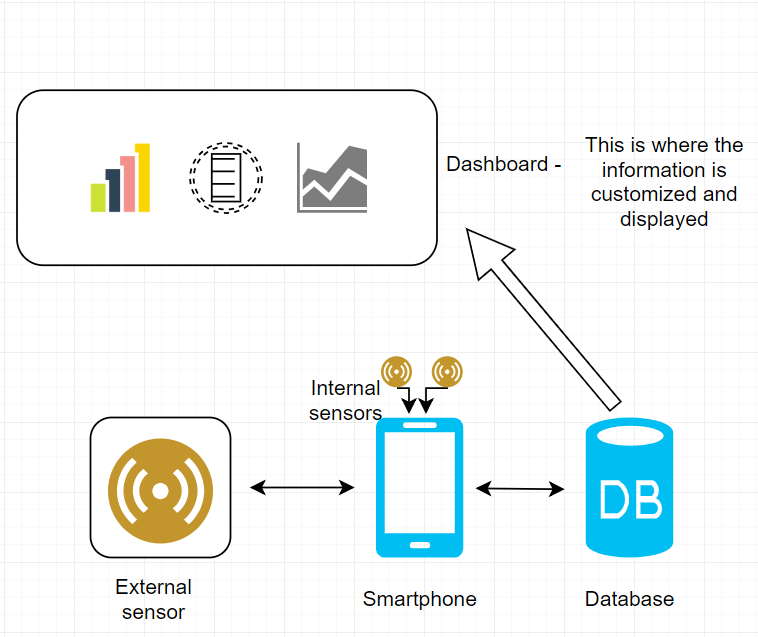
[Bluetooth 5](#_Toc17989699)

[Thingsboard 5](#_Toc17989700)

[Other Documentation 6](#_Toc17989701)

# Concept/design

## Basic diagram



NOTE:

Have a look at ‘Visualization of mapping.pptx’ (PowerPoint) for the brief explanation with images

## Platforms and tools

|  |  |  |  |
| --- | --- | --- | --- |
| **Data Collection**  (Sensors & Platform) |  | Middleware  (To process the data) | Visualization  (Display the data collected) |
| **SensorTag:**  - Accelerometer  - Gyroscope  - Humidity  - Light  - Pressure  - Temperature |  | Smartphone App (via BT)  Windows 8 App (via BT)  Linux App (via Rasp Pi) | Things board  Grafana Labs  A custom build  Google Edu  UMI  (AR & VR) |
| **Other Sensors:**  - Motion  - Weight  - Sound  - PIR |  |  |  |
| **Platforms**:  - Udoo  - Raspberry Pi  - Arduino  - Smartphone |  |  |  |
| **Others:**  - N Blocks  - Smart Meters:  Electricity , water etc. |  |  |  |

# List of extra features extended from the base ‘Google Science Journal’ app:

All images are saved in the ‘screenshots’ folder.

## Connection setup windows.

### Window one

(app\_screenshots folder - connection\_setup\_address.jpg)

1. User enters the web address of the database.

This is completed on the first run of the application and can be changed at any point via a menu option in the Main Activity.

* 1. Option one: Default. This is our ‘Thingsboard.tec-gateway.com’ address
  2. Option two: Custom. The user must have already created this account with Thingsboard.com

### Window two

(connection\_setup\_access\_token.jpg)

1. User enters experiment details.

Each experiment will have its own access token relating to a ‘device’ on the Thingsboard website. This way insures that data from each experiment will be kept separate. Making it easier to track and easier to create a dashboard to display the data.

* 1. User enters the device access Token
  2. User selects the connection type. MQTT (as fast as 100ms)/HTTP (Standard)

### Window three

(app\_screenshots folder - sensor\_frequency\_change.jpg)

1. User can select the frequency to send data.

This frequency can be unique to each sensor. This window is accessed via a menu option within each active sensor ‘display card’.

* 1. The sensor name will be displayed and the user will be able to select the ‘unit type’. Depending on the connection the user will have an option of:

Milliseconds/ seconds/ minutes/ hours.

* 1. A number picker wheel will be displayed and the user can select the chosen time frequency.

All setting are stored in the app and can be easily changed if needed.

## Bluetooth

(app\_screenshots folder - blustooth\_connection\_screen.jpg & blustooth\_connection\_screen\_connect.jpg)

1. Connection to SensorTag via Bluetooth.

The Bluetooth option that was original in place was design only to connect to a specified sensor type that was partnered with Google. This functionality was removed and a general device search was implemented. This code can search and display all BT devices within range, but currently only displays found SensorTags devices.

Future versions will have the ability to connect with other hardware i.e. Arduino/ Raspberry pi etc.

* 1. Search for Bluetooth devices and connect (current set to only display SensorTag devices).
  2. These SensorTag sensors will be displayed in the list of available sensors to monitor.
  3. Clicking on the sensor tab, or starting a ‘display card’ will start the process of charting the values and sending them periodically to Thingsboard.

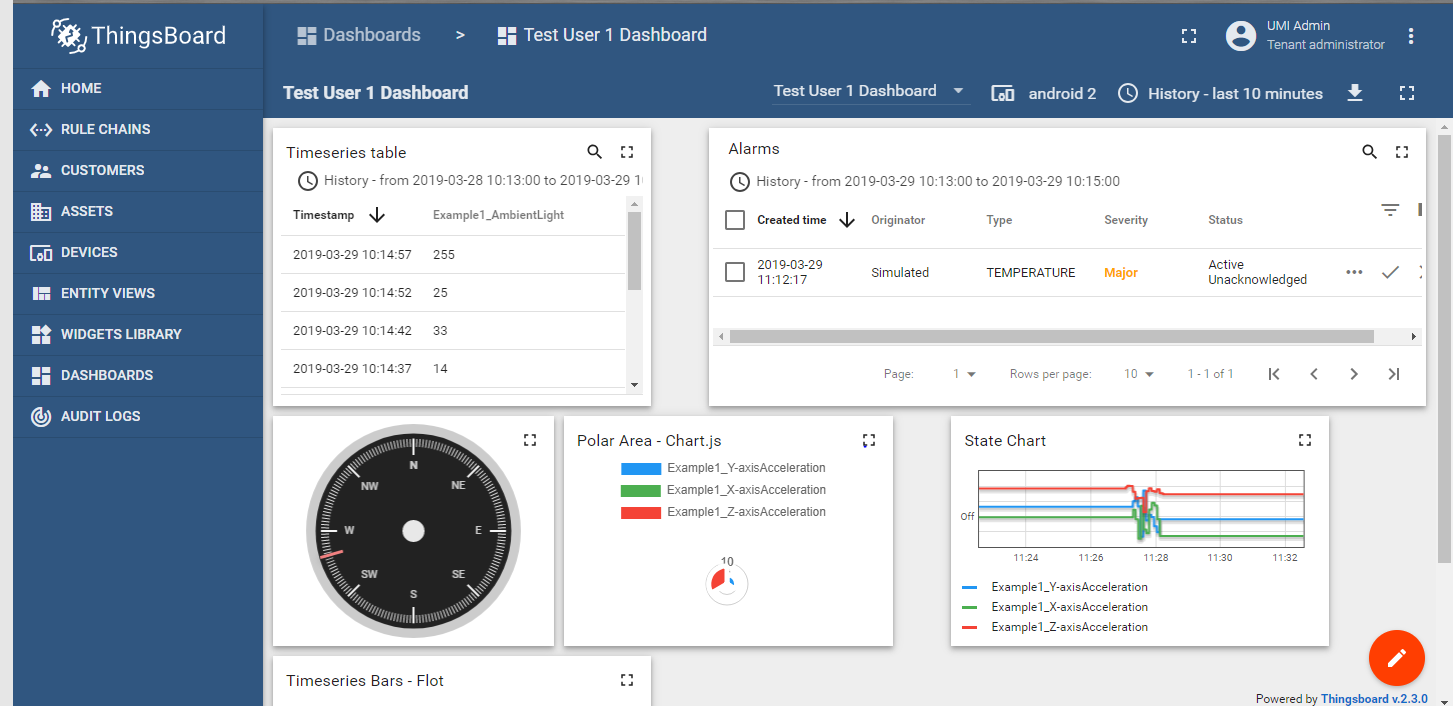
## Thingsboard

(ref – using\_thingsboard.doc)

1. Creating the devices and the dashboard widgets.

Within Thingsboard it is possible to create new ‘devices’ (sources of data), and with this ‘device’ see the data listed by name and to convert this stored data source to widgets that can be displayed visually.

* 1. Create a ‘device’ in Thingsboard, check the access token, and use this in the app when you create a new experiment.
  2. The data is sent to a Thingsboard database (our default or custom).
  3. Enter the device, select the data you wish to chart.
  4. Select the chart type you wish to use. Bar chart graph gauge etc.
  5. These dashboards can be modified even more after creation.



# Other Documentation

* user\_guide.doc - This is the app as a whole, the basic functionality with Nimbus modifications. This also goes through the use of Thingsboard
* visualization\_of\_mapping.pptx – Brief explanation of the app and Thingsboard.

# Issues

* 1. The tec gateway’s Thingsboard was move from the old server when they were replaced. Where is this now? Do we need to recreate… by we, I mean Gary
* 2. The BLE connect to SensorTags functionality is extremely glitchy. The code will search, and find the SensorTag. You can connect but then it gets messy. The integration of the sensors with the internal sensors is the issue. Sometimes it works fine, most of the time there is an overlap of data and two sensors are displayed on the one chart (image below) , or data from one sensor is displayed on the four charts for the external sensors
* 3. This is a small issue when the user enters a title for the experiment. As the user clicks on the first character of the title the ‘Untitled Experiment’ text should be removed leaving only the user’s input. i.e - ‘Untitled ExperimentNewName’… should be ‘NewName’

This image shows the Magnetometer (phone sensor) with two sources of data (one from an external sensor). This create the a large block of blue as the chart gets a low read and a high reading at the same time, continuously.



Low reading High reading