**Vetscan QR Testing**

**Table of Contents**

[1 Overview 2](#_Toc78206124)

[1.1 Scope of this Document 2](#_Toc78206125)

[1.2 Test Equipment 2](#_Toc78206126)

[1.3 Documentation 2](#_Toc78206127)

[1.4 QR Specification 3](#_Toc78206128)

[1.4.1 Content of QR code 3](#_Toc78206129)

[1.5 FUSE VETXML Test names 4](#_Toc78206130)

[1.6 Test Labels 5](#_Toc78206131)

[1.7 Test Label Generation 5](#_Toc78206132)

[2 Test #1: 6](#_Toc78206133)

[2.1 Test Setup 6](#_Toc78206134)

[3 Web App Test Instructions 7](#_Toc78206135)

[4 Test Results 8](#_Toc78206136)

[4.1 Vetscan Hub Mockup Web App 8](#_Toc78206137)

[4.2 Galaxy S10 Cell Phone 9](#_Toc78206138)

[4.3 Laptop Web App 10](#_Toc78206139)

[4.4 Laptop Python App 11](#_Toc78206140)

[4.5 Vetscan Hub Mockup Python App 12](#_Toc78206141)

[5 Conclusions 13](#_Toc78206142)

[6 Revision History 14](#_Toc78206143)

# Overview

This document will contain a test plan and report involving the Vetscan’s camera. This will mainly focus on the Vetscan’s camera ability to read QR codes.

Each combination of label size, module size, and total characters will be tested. Each test will be recorded as pass or fail.

## Scope of this Document

This document will be used for engineering staff to plan and document testing.

## Test Equipment

* Vetscan Hub mockup unit
* Ruler used to measure distance from the camera to a QR code, and the size of the QR code label.
* Printed QR Labels. Encoded according to ISO 18004:2006, using Level M error correction, and the max amount of data that can be contained in a QR Label.
* One carboard box – 1.7 inches tall and wider than 50 mm.
* One carboard box – 3.5 inches tall and wider than 50 mm.
* qr\_code\_genertor.py – A Python application used to generate QR labels.

## Documentation

|  |  |
| --- | --- |
| ISO 18004:2006 | QR code specification |
| <https://www.the-qrcode-generator.com/>scan | Web site – create or read a QR label with given data |

## QR Specification

### Content of QR code

**Format:** the content of the QR code must be a valid json structure.

**Name:** Mandatory information. Valid names are current FUSE VETXML Test codes.

See Name table below.

Maximum length 8 characters.

**Lot:** Mandatory information. A string consistent with the lot id from the manufacturer.

**Exp:** Mandatory information. Expiration date expressed according to ISO 8601 as yyyy-mm-dd.

**Data:** Optional. String of data to be sent verbatim to the analyzer.

Valid QR code according to specification (ISO 18004:2006)

Version: Minimum 1 (21 × 21 modules)  
 Maximum 40 (177 × 177 modules)

Size: Minimum size 10 mm × 10 mm.  
 Maximum size 50 mm × 50 mm.

Error correction: Level M.

Printed Labels: printed on a high-contrast background

Example JSON string that would be encoded into a QR label.

{

"analyzer": "<analyzer id>",

"uuid": "<uuid>",

"lot": "<string>",

"exp": "<yyyy-mm-dd>",

"data": "<optional data>"

}

This is 53 characters long without any data or whitespace, just the name value pairs and syntax requirements. An example SPE QR would be as follows:

{

"analyzer":"vetscan-spe",

"exp":"2022-05-07",

"uuid":"0b7ec890-3960-11eb-a081-2790e47ff2f4",

"lot":"1234",

"data":""

}

This is 114 characters when whitespace is removed.

## FUSE VETXML Test names

|  |  |  |
| --- | --- | --- |
| **Analyzer** | **Name** | **Description** |
| Chemistry | AR | Avian/Reptilian Profile Plus |
| CDP | Comprehensive Diagnostic |
| CC | Critical Care Plus |
| EPP | Equine Profile Plus |
| KPP | Kidney Profile Plus |
| LA | Large Animal Profile |
| MLP | Mammalian Liver Profile |
| Prep | Prep Profile II |
| T4 | Thyroxine(T4)/Cholesterol Test |
| EP | Electrolyte Plus |
| PCP | Preventive Care Profile Plus |
| PP | Phenobarbital Profile |
| Hematology | HEM | Hematology |
| Chemistry2 | PT/aPTT | PT/aPTT Combination Test |
| Fib | Equine Fibrinogen Test |
| EFib | Equine Fibrinogen |
| CFib | Canine Fibrinogen |
| CBT | Canine Blood Typing |
| FBT | Feline Blood Typing |
| PTaPTT | PT/aPTT Combination |
| Phb | Phenobarbital (not released) |
| RapidTests | E | Ehrlichia Rapid Test |
| P | Parvo Rapid Test |
| A | Anaplasma Rapid Test |
| G | Giardia Rapid Test |
| L | Lyme Rapid Test |
| FF | FeLV\_Fiv Rapid Test |
| c | cPL Rapid Test |
| FLE | Flex4 Rapid Test |
| H | Heartworm Rapid Test |
| Fecalanalysis | FOVA | Fecal Ova/Oocysts |
| FGIA | Fecal Giardia |
| Urinalysis | SA | Urine Sediment |
| SA10 | Urine Sediment and UA10 |
| SA14 | Urine Sediment and UA14 |
| UA10 | UA10 |
| UA14 | UA14 |

## Test Labels

A QR code (abbreviated from Quick Response code) is a type of matrix barcode (or two-dimensional barcode[1]) invented in 1994 by the Japanese automotive company Denso Wave.[2] A barcode is a machine-readable optical label that contains information about the item to which it is attached. In practice, QR codes often contain data for a locator, identifier, or tracker that points to a website or application. A QR code uses four standardized encoding modes (numeric, alphanumeric, byte/binary, and kanji) to store data efficiently; extensions may also be used.

* The test labels will be printed on paper with matte and glossy finishes.
* The test labels will be scanned on boxes that are 1.7 inches and 3.5 inches tall.
* The test Label Sizes that will be used will be 10 mm x 10 mm and 50 x 50 mm.
* The test Labels will contain a range of test character strings.

## Test Label Generation

* The qr\_code\_generator.py Python application was used to generate the images for the different label data densities.
  + https://github.com/lincolnloop/python-qrcode
* The app created image files in the QR\_labels directory.
* The QR\_labels/QR\_labels.docx file contains tables for containing the QR label images.
* The images in the table were labeled with the file name, which contains the QR code version and number of characters in the QR label.
* The QR label images were pasted into the tables in the QR\_labels/QR\_labels.docx file.
* Images in the tables were sized to 50mm x 50mm, 20mm x 20mm, 10mm x 10mm.
* The QR\_labels.docx file was printed at high resolution on a laser printer in black and white on stock white paper.
* The individual QR labels were cut out of the printed paper.

## Test Label Printing

Using the program ifranview (<https://www.irfanview.com/>), it is possible to scale and print all the QR codes at once.

1. Install irfanview
2. Open irfanview
3. Select File->Thumbnails
4. Using file view pane on left hand side of screen, navigate to the directory where the QR code images are located
5. Now click on the files to select the ones you wish to print
6. Select File->Print selected files as single images (batch print)
7. Set Print Size to custom and set as desired size in cm in Width and Height input boxes. The cm setting is set in Units for ‘custom’ and ‘position’ found below. Can also be set to images.
8. Now check the box for Headnote and put $F in the input box so that the file name is printed with the image
9. Select Center horz in the Position on paper section
10. You can go into the Printer setup and set the printer options on the printer driver (2 sided, flip, etc)

# Test #1:

The purpose of this test is to determine what is the maximum amount of data that can be read with the camera at given distances, label sizes, and label densities.

## Test Setup

* The web camera and QR labels will be tested using a web site that can activate the camera and record the QR label’s code. The web page will display the camera image and display the QR code.
* The QR labels will be placed on the top of ether the 1.7-inch or 3.5-inch-tall box.
* The box will be placed on the base of the Vetscan Hub Mockup.
* The display of the Vetscan Hub Mockup will need to be tilted until it aligns to the label.
* The web site will use the Vetscan Hub Mockup’s camera to capture an image of the QR label and attempt to decode the embedded data in the QR label.
* If the QR label is decoded, the data will be displayed on the web page. Each test case shall be recorded in the tables below. If the QR label is decoded, then the test case shall be marked as “Pass”, else it shall be marked as “Fail”.

See screen capture of web page below.



# Test Instructions

* For 10 x 10 mm and 20 x 20 mm, use a 3.5” box to elevate the QR code off the base of the Vetscan Hub Mockup
* For 50 x 50 mm, use a 1.5” box to elevate the QR code off the base of the Vetscan Hub Mockup

## Web App

Perform the following test setup once at the beginning of testing:

1. Turn on the Vetscan Hub Mockup or Laptop that is under test.
2. Open the web site: <https://4qrcode.com/scan-qr-code.php>.
3. On the web page, click on the button labelled “Open camera”.
4. You should see an image from the unit’s camera displayed on the web page.

For each test case, select the proper box and QR label.

1. Place the test box under the Vetscan Hub Mockup’s or Laptop’s camera.
2. Place the QR label on the top of the box if performing test on Vetscan Hub Mockup.
3. Using the web page’s camera image, center the QR tag’s image in the center of the image.
4. If the QR label can be decoded, the web page will display the QR label’s data on web page.
5. Once the image is centered, wait at most 30 seconds for the web site to decode the QR label.
6. If the QR label was decoded within 30 seconds, then the test case passed, else the test case failed.
7. Record the pass/fail status of each test case.

## Galaxy S10

Perform the following test setup once at the beginning of testing:

1. Open the QR & Barcode Reader – TeaCapps on the Galaxy S10.

For each test case, select the proper box and QR label:

1. Place the test box under the Galaxy S10’s camera.
2. Using the camera view on the phone, center the QR tag’s image in the center of the image.
3. If the QR label can be decoded, the value will be displayed on the phone screen.
4. If the QR was decoded within 30 seconds, then the testcase passes, else the test case failed.
5. Record the pass/status of each test case.

## Python App

Perform the following test setup once at the beginning of testing:

1. Turn on the Vetscan Hub Mockup or the Laptop being tested.

For each test case, select the proper box and QR label.

1. Place the test box under the Vetscan Hub Mockup’s or Laptop’s camera.
2. Place the QR label on the top of the box if performing test on Vetscan Hub Mockup.
3. Initiate the qr scanner python app from the console.
4. Using the displayed camera image, center the QR tag’s image in the center of the image.
5. If the QR label can be decoded, the output will be displayed on the terminal.
6. Once the image is centered, wait at most 30 seconds for the web site to decode the QR label.
7. If the QR label was decoded within 30 seconds, then the test case passed, else the test case failed.
8. Record the pass/fail status of each test case.

## 

# Test Results

## Vetscan Hub Mockup Web App

* These test results were recorded on the Vetscan.
* Web page: https://www.the-qrcode-generator.com/scan

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Version** | **Modules** | **Max**  **Chars**  **(per spec)** | **Max**  **Chars**  **(base +**  **numbers)** | **(Pass/Fail)**  **10 x 10 mm** | **Pass/Fail**  **20 x 20 mm** | **Pass/Fail**  **50 x 50 mm** |
| 1 | 21 x 21 | 14 |  | Pass | Pass | Pass |
| 2 | 25 x 25 | 26 |  |  |  |  |
| 3 | 29 x 29 | 42 |  | Pass | Pass |  |
| 4 | 33 x 33 | 62 |  | Pass | Pass |  |
| 5 | 37 x 37 | 84 | 104 | Fail | Pass |  |
| 6 | 41 x 41 | 106 | 156 |  | Pass |  |
| 7 | 45 x 45 | 122 | 195 |  | Pass |  |
| 8 | 49 x 49 | 152 | 267 |  | Pass |  |
| 9 | 53 x 53 | 180 | 334 |  | Pass |  |
| 10 | 57 x 57 | 213 | 410 |  | Pass | Pass |
| 11 | 61 x 61 | 251 | 501 |  | Pass |  |
| 12 | 65 x 65 | 287 | 588 |  | Pass | Pass |
| 13 | 69 x 69 | 331 | 693 |  | Fail |  |
| 14 | 73 x 73 | 362 | 768 |  | Fail | Pass |
| 15 | 77 x 77 | 412 | 888 |  | Fail | Fail |
| 16 | 81 x 81 | 450 | 979 |  |  | Fail |
| 17 | 85 x 85 | 504 | 1109 |  |  |  |
| 18 | 89 x 89 | 560 | 1243 |  |  |  |
| 19 | 93 x 93 | 624 | 1379 |  |  |  |
| 20 | 97 x 97 | 666 | 1497 |  |  |  |
| 21 | 101 x 101 | 711 | 1605 |  |  |  |
| 22 | 105 x 105 | 779 | 1769 |  |  |  |
| 23 | 109 x 109 | 857 | 1956 |  |  |  |
| 24 | 113 x 113 | 911 | 2085 |  |  |  |
| 25 | 117 x 117 | 997 |  |  |  |  |
| 26 | 121 x 121 | 1059 |  |  |  |  |
| 30 | 137 x 137 | 1370 |  |  |  |  |
| 31 | 141 x 141 | 1452 |  |  |  |  |
| 32 | 145 x 145 | 1538 |  |  |  |  |
| 33 | 149 x 149 | 1628 |  |  |  |  |
| 34 | 153 x 153 | 1722 |  |  |  |  |
| 35 | 157 x 157 | 1809 |  |  |  |  |
| 40 | 177 x 177 | 2331 |  |  |  |  |

## Galaxy S10 Cell Phone

* These test results were recorded on a Galaxy S10 cell phone.
* Andriod app: QR & Barcode Reader - TeaCapps
* Distance from camera: ~2-inches

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Version** | **Modules** | **Max Characters** | **(Pass/Fail)**  **10 x 10 mm** | **Pass/Fail**  **20 x 20 mm** | **Pass/Fail**  **50 x 50 mm** |
| 1 | 21 x 21 | 14 | Pass | Pass | Pass |
| 2 | 25 x 25 | 38/26 |  | Pass |  |
| 3 | 29 x 29 | 53/42 | Pass | Pass |  |
| 4 | 33 x 33 | 82/62 | Pass | Pass |  |
| 5 | 37 x 37 | 113/84 | Pass | Pass |  |
| 6 | 41 x 41 | 145/106 | Pass | Pass |  |
| 7 | 45 x 45 | 169/122 | Pass | Pass |  |
| 8 | 49 x 49 | 212/152 | Pass | Pass |  |
| 9 | 53 x 53 | 253/180 | Fail | Pass |  |
| 10 | 57 x 57 | 301/213 | Fail | Pass | Pass |
| 11 | 61 x 61 | 419/251 |  | Pass |  |
| 12 | 65 x 65 | 408/287 |  | Pass | Pass |
| 13 | 69 x 69 | 472/331 |  | Pass |  |
| 14 | 73 x 73 | 517/362 |  | Pass | Pass |
| 15 | 77 x 77 | 590/412 |  | Pass | Pass |
| 16 | 81 x 81 | 600/450 |  | Pass |  |
| 17 | 85 x 85 | 656/504 |  | Pass |  |
| 18 | 89 x 89 | 734/560 |  | Pass |  |
| 19 | 93 x 93 | 816/624 |  | Pass |  |
| 20 | 97 x 97 | 909/666 |  | Pass | Pass |
| 21 | 101 x 101 | 970/711 |  | Pass |  |
| 22 | 105 x 105 | 1035/779 |  | Pass |  |
| 23 | 109 x 109 | 1134/857 |  | Pass |  |
| 24 | 113 x 113 | 1248/911 |  | Fail |  |
| 25 | 117 x 117 | 1326/997 |  | Fail |  |
| 26 | 121 x 121 | 1451/1059 |  | Fail |  |
| 30 | 137 x 137 | 1983/1370 |  |  | Pass |
| 31 | 141 x 141 | 2102/1452 |  |  | Fail |
| 32 | 145 x 145 | 2227/1538 |  |  | Fail |
| 33 | 149 x 149 | 2358/1628 |  |  | Fail |
| 34 | 153 x 153 | 2495/1722 |  |  | Fail |
| 35 | 157 x 157 | 2621/1809 |  |  | Fail |
| 40 | 177 x 177 | 3380/2331 |  |  | Fail |

## Laptop Web App

* These test results were recorded on a ThinkPad P50.
* Web page: https://www.the-qrcode-generator.com
* Distance from Web camera: ~2.25-inches

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Version** | **Modules** | **Max Characters** | **(Pass/Fail)**  **10 x 10 mm** | **Pass/Fail**  **20 x 20 mm** | **Pass/Fail**  **50 x 50 mm** |
| 1 | 21 x 21 | 14 |  | Pass |  |
| 2 | 25 x 25 | 38/26 |  | Pass |  |
| 3 | 29 x 29 | 53/42 |  | Pass |  |
| 4 | 33 x 33 | 82/62 |  | Pass |  |
| 5 | 37 x 37 | 113/84 |  | Pass |  |
| 6 | 41 x 41 | 145/106 |  | Pass |  |
| 7 | 45 x 45 | 169/122 |  | Pass |  |
| 8 | 49 x 49 | 212/152 |  | Pass |  |
| 9 | 53 x 53 | 253/180 |  | Fail |  |
| 10 | 57 x 57 | 301/213 |  | Fail |  |
| 11 | 61 x 61 | 419/251 |  | Fail |  |
| 12 | 65 x 65 | 408/287 |  | Fail |  |
| 13 | 69 x 69 | 472/331 |  | Fail |  |
| 14 | 73 x 73 | 517/362 |  | Fail |  |
| 15 | 77 x 77 | 590/412 |  | Fail |  |
| 16 | 81 x 81 | 600/450 |  |  |  |
| 17 | 85 x 85 | 656/504 |  |  |  |
| 18 | 89 x 89 | 734/560 |  |  |  |
| 19 | 93 x 93 | 816/624 |  |  |  |
| 20 | 97 x 97 | 909/666 |  |  |  |
| 21 | 101 x 101 | 970/711 |  |  |  |
| 22 | 105 x 105 | 1035/779 |  |  |  |
| 23 | 109 x 109 | 1134/857 |  |  |  |
| 24 | 113 x 113 | 1248/911 |  |  |  |
| 25 | 117 x 117 | 1326/997 |  |  |  |
| 26 | 121 x 121 | 1451/1059 |  |  |  |
| 30 | 137 x 137 | 1983/1370 |  |  |  |
| 31 | 141 x 141 | 2102/1452 |  |  |  |
| 32 | 145 x 145 | 2227/1538 |  |  |  |
| 33 | 149 x 149 | 2358/1628 |  |  |  |
| 34 | 153 x 153 | 2495/1722 |  |  |  |
| 35 | 157 x 157 | 2621/1809 |  |  |  |
| 40 | 177 x 177 | 3380/2331 |  |  |  |

## Laptop Python App

* These test results were recorded on a ThinkPad P50.
* Python app: qr\_scanner.py
* Distance from Web camera: ~2.25-inches

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Version** | **Modules** | **Max Characters** | **(Pass/Fail)**  **10 x 10 mm** | **Pass/Fail**  **20 x 20 mm** | **Pass/Fail**  **50 x 50 mm** |
| 1 | 21 x 21 | 14 |  | Pass |  |
| 2 | 25 x 25 | 38/26 |  | Pass |  |
| 3 | 29 x 29 | 53/42 |  | Pass |  |
| 4 | 33 x 33 | 82/62 |  | Pass |  |
| 5 | 37 x 37 | 113/84 |  | Pass |  |
| 6 | 41 x 41 | 145/106 |  | Pass |  |
| 7 | 45 x 45 | 169/122 |  | Pass |  |
| 8 | 49 x 49 | 212/152 |  | Pass |  |
| 9 | 53 x 53 | 253/180 |  | Pass |  |
| 10 | 57 x 57 | 301/213 |  | Fail |  |
| 11 | 61 x 61 | 419/251 |  | Pass |  |
| 12 | 65 x 65 | 408/287 |  | Fail |  |
| 13 | 69 x 69 | 472/331 |  | Fail |  |
| 14 | 73 x 73 | 517/362 |  |  |  |
| 15 | 77 x 77 | 590/412 |  |  |  |
| 16 | 81 x 81 | 600/450 |  |  |  |
| 17 | 85 x 85 | 656/504 |  |  |  |
| 18 | 89 x 89 | 734/560 |  |  |  |
| 19 | 93 x 93 | 816/624 |  |  |  |
| 20 | 97 x 97 | 909/666 |  |  |  |
| 21 | 101 x 101 | 970/711 |  |  |  |
| 22 | 105 x 105 | 1035/779 |  |  |  |
| 23 | 109 x 109 | 1134/857 |  |  |  |
| 24 | 113 x 113 | 1248/911 |  |  |  |
| 25 | 117 x 117 | 1326/997 |  |  |  |
| 26 | 121 x 121 | 1451/1059 |  |  |  |
| 30 | 137 x 137 | 1983/1370 |  |  |  |
| 31 | 141 x 141 | 2102/1452 |  |  |  |
| 32 | 145 x 145 | 2227/1538 |  |  |  |
| 33 | 149 x 149 | 2358/1628 |  |  |  |
| 34 | 153 x 153 | 2495/1722 |  |  |  |
| 35 | 157 x 157 | 2621/1809 |  |  |  |
| 40 | 177 x 177 | 3380/2331 |  |  |  |

## Vetscan Hub Mockup Python App

* These test results were recorded on the Vetscan.
* Python app: qr\_scanner.py

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Version** | **Modules** | **Max Characters** | **(Pass/Fail)**  **10 x 10 mm** | **Pass/Fail**  **20 x 20 mm** | **Pass/Fail**  **50 x 50 mm** |
| 1 | 21 x 21 | 14 |  | Pass |  |
| 2 | 25 x 25 | 38/26 |  |  |  |
| 3 | 29 x 29 | 53/42 |  | Pass |  |
| 4 | 33 x 33 | 82/62 |  | Pass |  |
| 5 | 37 x 37 | 113/84 |  | Pass |  |
| 6 | 41 x 41 | 145/106 |  | Pass |  |
| 7 | 45 x 45 | 169/122 |  | Pass |  |
| 8 | 49 x 49 | 212/152 |  | Pass |  |
| 9 | 53 x 53 | 253/180 |  | Pass |  |
| 10 | 57 x 57 | 301/213 |  | Pass |  |
| 11 | 61 x 61 | 419/251 |  | Pass |  |
| 12 | 65 x 65 | 408/287 |  | Pass |  |
| 13 | 69 x 69 | 472/331 |  | Pass |  |
| 14 | 73 x 73 | 517/362 |  | Pass |  |
| 15 | 77 x 77 | 590/412 |  | Pass |  |
| 16 | 81 x 81 | 600/450 |  | Pass |  |
| 17 | 85 x 85 | 656/504 |  | Fail |  |
| 18 | 89 x 89 | 734/560 |  | Fail |  |
| 19 | 93 x 93 | 816/624 |  | Fail |  |
| 20 | 97 x 97 | 909/666 |  | Fail |  |
| 21 | 101 x 101 | 970/711 |  |  |  |
| 22 | 105 x 105 | 1035/779 |  |  |  |
| 23 | 109 x 109 | 1134/857 |  |  |  |
| 24 | 113 x 113 | 1248/911 |  |  |  |
| 25 | 117 x 117 | 1326/997 |  |  |  |
| 26 | 121 x 121 | 1451/1059 |  |  |  |
| 30 | 137 x 137 | 1983/1370 |  |  |  |
| 31 | 141 x 141 | 2102/1452 |  |  |  |
| 32 | 145 x 145 | 2227/1538 |  |  |  |
| 33 | 149 x 149 | 2358/1628 |  |  |  |
| 34 | 153 x 153 | 2495/1722 |  |  |  |
| 35 | 157 x 157 | 2621/1809 |  |  |  |
| 40 | 177 x 177 | 3380/2331 |  |  |  |

# Conclusions

The glare from the Vetscan’s camera light is causing glare in the captured image. The QR reader is not able to read the label when the glare appears within the image of the QR label.

The current version of the Vetscan’s camera limits it to reading 10mm x 10mm QR labels that have 82 or less characters using Version 3 encoding.

The current version of the Vetscan’s camera limits it to reading 50mm x 50mm QR labels that have 517 or less characters using Version 14 encoding.

The Vetscan’s camera focal length of the camera is so far from the lens that the high density, 10mm x 10mm QR labels can’t be read. At about 1 inch from the camera, the image is very small.

The Python app max version was 16, for 600 max characters. The web app max version was 12, for 408 max characters. The Python app was 50% better at decoding the lables.

# Revision History

Record the results of testing labels at 3 inches (76.2 mm).

|  |  |  |  |
| --- | --- | --- | --- |
| **Version** | **Date** | **Author** | **Comments** |
| A beta | 15 JUL 2021 | Bruce Graham | Initial work. The use of the 10 mm x 10 mm QR labels is to be resolved with initial testing. |
| 1 | 22 JUL 2021 | Brian Newberry | 1.4.1 Updated Content of label code. |
| 2 | 26 JUL 2021 | Bruce Graham | 4.5 Added test results Vetscan: 20mm  4.6 Added test results Laptop: 20mm -web app  4.7 Added test results Laptop: 20mm – Python app  4.8 Added test results Galaxy S10 cell phone: 20mm  4.9 Added test results – Vetscan: 20mm with Python app |