RETINAS Test Plan

[**1. SCOPE**](#_tshndl7ol2d8) **2**

[**2. REASON FOR RE-ISSUE**](#_rbdilolecqvf) **2**

[**3. INTRODUCTION**](#_8d6adqsyi27d) **2**

[3.1. Description](#_72ozew50tcnb) 2

[3.2. Strategy](#_br31otcwda3w) 2

[**4. APPLICABLE DOCUMENT**](#_oh8l1dpzccvg) **2**

[**5. DEFINITIONS**](#_ai9dine6lrie) **3**

[**6. SETUP**](#_z58w7yv2zfk1) **5**

[**7. EQUIPMENT RECORD**](#_tnqqxr7a93ty) **5**

[**8. TEST CASES**](#_e6u8rrf9t3u5) **6**

[[Test Case 010] Database Connectivity (Login)](#_4gvl3b2yyia6) 6

[[Test Case 020] RETINAS Server Data Upload](#_9kc5nn5ijc1e) 8

[[Test Case 030] load data program](#_fgtxvj4p01yx) 12

[[Test Case 040] Proper classification, artifact removal, and metrics for mixed data](#_nije4ow6r3qa) 14

[**9. ATTACHMENTS**](#_t6x56iteyp03) **17**

# 

# **1.** **SCOPE**

This document describes the test plan for verifying the design of the RETINAS as described in Requirement documents. It provides the necessary procedures for the validation test of the device.

# 2. REASON FOR RE-ISSUE

**ISSUE REASON FOR RE-ISSUE**

1 MR #RETINAS001 This is the first time the plan has been issued

# **3.** **INTRODUCTION**

## 3.1. Description

The following tests will be performed:

|  |  |  |
| --- | --- | --- |
| **Test Case** | **Direct Requirement** | **Test** |
| 010 | REQ010, REQ011, REQ080, REQ081, REQ400, REQ410REQ010 & REQ011 | Database Connectivity |
| 020 | REQ 30- REQ50 | RETINAS Server Data Upload |
| 030 | REQ 52 | Select Raw Data |
| 040 | REQ50 - REQ 370 | Mixed Data Selected and Analyzed |

## 3.2. Strategy

1. RETINAS is a new product. All tests will be conducted.

# 4. APPLICABLE DOCUMENT

This plan is based on requirements from RETINAS001 Software requirements.

# 

# 

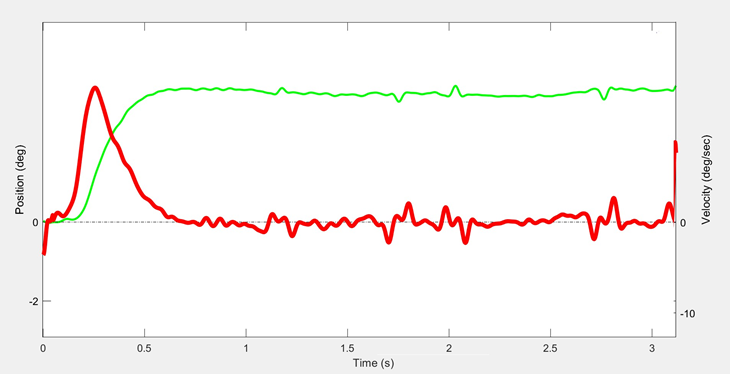
# 5. DEFINITIONS

**RETINAS** - Real-Time Eye Tracking Impartial Numerical Analysis Software

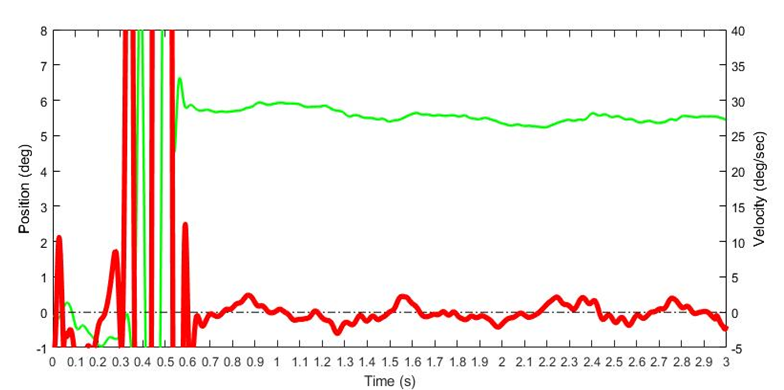
**CSV** - Comma Separated Values - a text file saved in a tabulated comma-separated form that can be opened in excel and text editors

**Gold Standard Values** - golden standard for data classification and validation are eye movements analyzed by human operators already contained in the database and available at separate CSVs

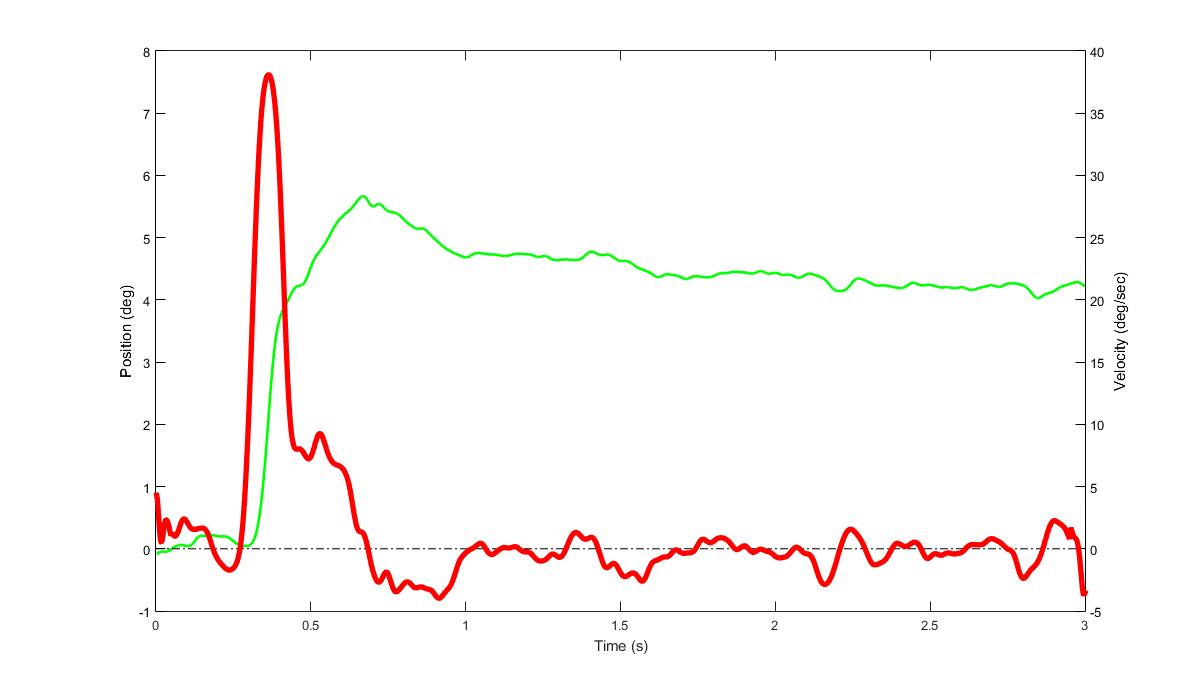
**Good:**



**BAD:**



**MAYBE:**



Maybe files can be fixable, where it is simple to remove the artifacts, or unfixable, where fixing them can alter the metric values so should not be fixed so any metrics cannot be retrieved

# **6.** **SETUP**

The testing requires the following units as a minimum: RETINAS software application and Internet Access

# 7. EQUIPMENT RECORD

This testing requires the following units as a minimum: A computer with access to the RETINAS software

The following test equipment, or equivalent, is needed to execute the tests in this plan.

|  |  |  |
| --- | --- | --- |
| **Item** | **Model Number(s)** | **Calibration Required?** |
| Computer | DELL | No |
| RETINAS Software | Eclipse with Python compatible environment | No |
| Ethernet Connection | Internet Service Provider Dependent | No |
| Database Domain | seapickle.ftp.sh /seapickle.ftp.sh/phpmyadmin | No |
| rawfiles Folder |  | No |
| Eclipse IDE | Version 2018-09 (4.9.0) | No |
| Pydev | Version 7.5.0 | No |
| Previously Analyzed Eye Movement Data |  | No |
| Eye Movements Data |  | No |
| RETINAS Database | seapickle.ftp.sh/phpmyadmin | No |
| Web Browser | Google Chrome, Firefox, Edge, Safari | No |

# 

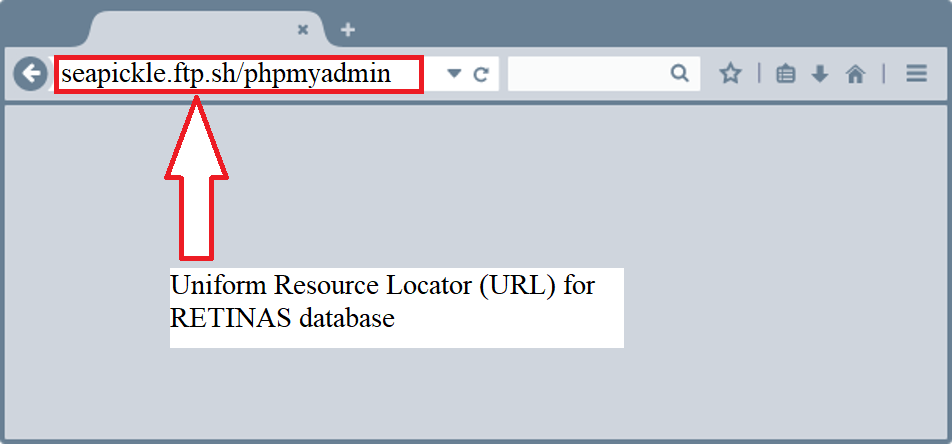
# 8. TEST CASES

### [Test Case 010] Database Connectivity (Login)

**Purpose**: To verify that RETINAS database is able to be logged into (connected to) via an internet (web) browser

**Specification:** REQ010, REQ011, REQ080, REQ081, REQ400, REQ410REQ010 & REQ011

**Test Architecture:**

****

[**https://www.clipart.email/clipart/internet-browser-clipart-93587.html**](https://www.clipart.email/clipart/internet-browser-clipart-93587.html)

**Figure 1**a *An internet connected web-browser with the URL address for the RETINAS database login page.*

**Equipment:**

Computer with Windows or Mac Operating System

Wifi/Ethernet (Web)Access

**Test Procedure**:

1. Type “<http://seapickle.ftp.sh/phpmyadmin/>” in a web browser’s URL (Uniform Resource Locator) address bar. (See Figure 1a)
2. View the RETINAS phpmyadmin login screen. (Refer to Figure 1.b)

****

**Figure 1b** *The phpMyAdmin login page for the RETINAS database*

**Expected Results:**

1. The RETINAS database URL is active and able to be reached via login through an internet-capable web browser according to procedure #1.
2. The RETINAS database phpmyadmin page appears when connection is successful according to procedure #2.

**Test Results:**

|  |  |
| --- | --- |
| **Item** | **Pass/Fail** |
| The RETINAS database URL is active and able to be reached via a internet-capable web browser according to procedure #1. | Pass |
| The RETINAS database phpmyadmin page appears when connection is successful according to procedure #2. | Pass |

### 

### [Test Case 020] RETINAS Server Data Upload

**Purpose:** Verify data upload to RETINAS server

**Specification:** REQ031, REQ032

**Test Architecture:**

1. Download Data Parsing Script from [here](https://drive.google.com/open?id=1gp-Y3lvhP8TAwYIYk16Y1gsey0YQmNjI)
2. Open Eclipse
   1. Click “File”
   2. Select “New”
      1. Select “Pydev Project”
      2. Name project “Test”
3. Download “rawfiles” Folder from [here](https://drive.google.com/open?id=1T-iiYraV1ZdPnO4CahElYOIIuoewI2KN)
4. Modify the Eclipse directory
   1. Go to “File” in the upper left corner
   2. Hover over “Switch Workspace”
   3. Click “Other”
   4. Insert: “C:\Users\USERNAME\Downloads\rawfiles”
      1. USERNAME is specified by the computer User’s account’s name

**Equipment:**

seapickle.ftp.sh/phpmyadmin

Computer with python 3.7 compatible environment and mysqlconnector installed

Wifi/Ethernet Connection

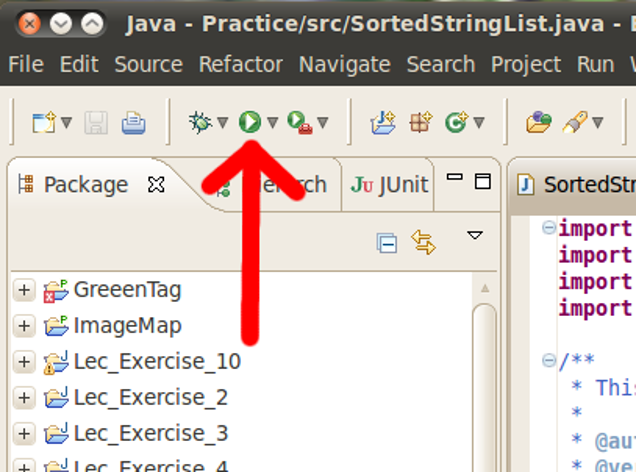
Web browser (such as Google Chrome version 33.0.1750 or higher)

Data Parsing Script **LINK:** [**here**](https://drive.google.com/open?id=1gp-Y3lvhP8TAwYIYk16Y1gsey0YQmNjI)

Test Data File [**here**](https://drive.google.com/open?id=1T-iiYraV1ZdPnO4CahElYOIIuoewI2KN)

**Test Procedure:**

1. Open Data Parsing Script in Eclipse Pydev environment
2. Open the “rawfiles” Folder
3. Run the Data Parsing Script by pressing green arrow (Figure 1c)



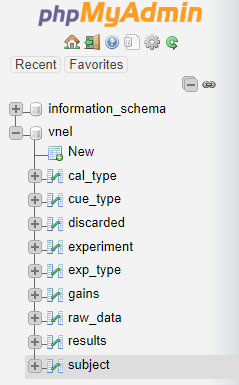
**Figure 1c** The green arrow shown provides how the Eclipse software executes the Data Parsing Script.

1. Log into seapickle.ftp.sh/phpmyadmin (See Figure1d)
   1. Provide username in “vnel” field
   2. Provide password in “ana14sis” field

****

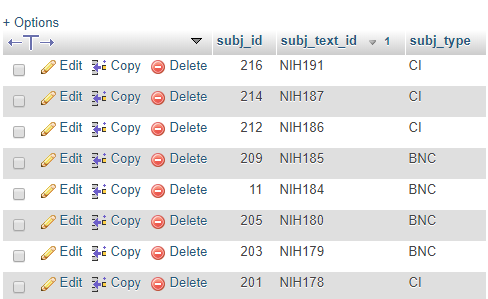
**Figure 1d** Depicts the login page to “seapickle.ftp.sh/phpmyadmin”

1. Find subject table (Figure 2)



**Figure 2** Provides the table listings for the RETINAS database, seen at the bottom “subject” table is highlighted.

1. Locate subj\_text\_id (See Figure 3)



**Figure 3** From selecting the “subject” table, as depicted above the second column provided is labeled “subj\_text\_id”

1. Locate within subj\_text\_id “NIH999” (Figure 4)



**Figure 4** Shown above, utilizing the “subj\_text\_id”, test subject “NIH999” can be identified.

**Expected Results:**

1. “NIH999” is located within the RETINAS database under the “subjects” table within column “subj\_text\_id”

**Test Results:**

|  |  |
| --- | --- |
| **Item** | **Pass/Fail** |
| “NIH999” is located within seapickle.ftp.sh/phpmyadmin under “subj\_text\_id” | Pass |

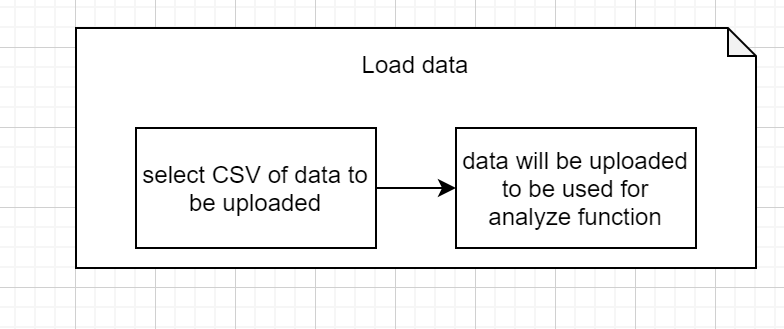
### 

### [Test Case 030] load data program

**Purpose:** To verify that the Raw Data can be selected using the program

**Specification:** REQ 52

**Test Architecture:**



**Figure 1** *load function flowchart*

**Equipment:**

1. Computer with RETINAS installed
2. Internet connection
3. Raw eye movement data filehttps://drive.google.com/open?id=1MjTsVpMIO7kVZfBj1xarF42vQ6iLP\_nQ
   1. Data file consists of 5 different files that have different scenarios.
      1. Scenario I : 1 good eye data file (“good”)

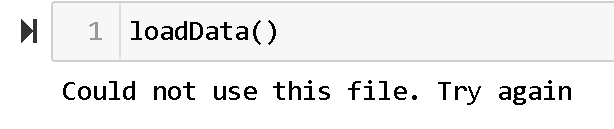
**Test Procedure**:

1. Turn on computer
2. Open Jupyter notebook environment
3. In Jupyter environment, open RETINAS folder
4. Click “select raw data”
5. Navigate to the Desktop in the popup window and select “rawData” folder that was previously created
6. Do not click on any data file
7. Click select
8. **\*check for the error message**
9. Close error message
10. Click on the “good” file
11. Click Select

**Expected Results**

Select Raw Data operation is operating properly if the following conditions are true:

1. Step 8: When no file is selected, A message displaying “No files were selected for processing ” displays



**Figure 2:** Error Message when no files are selected

1. When file is selected in step 11, A message displaying “File Successfully Stored” displays

**Test Results:**

|  |  |
| --- | --- |
| **Item** | **Pass/Fail** |
| Message displaying “Could not use this file. Try again ” | Pass |
| Message displaying “File Successfully Stored” | Pass |

### 

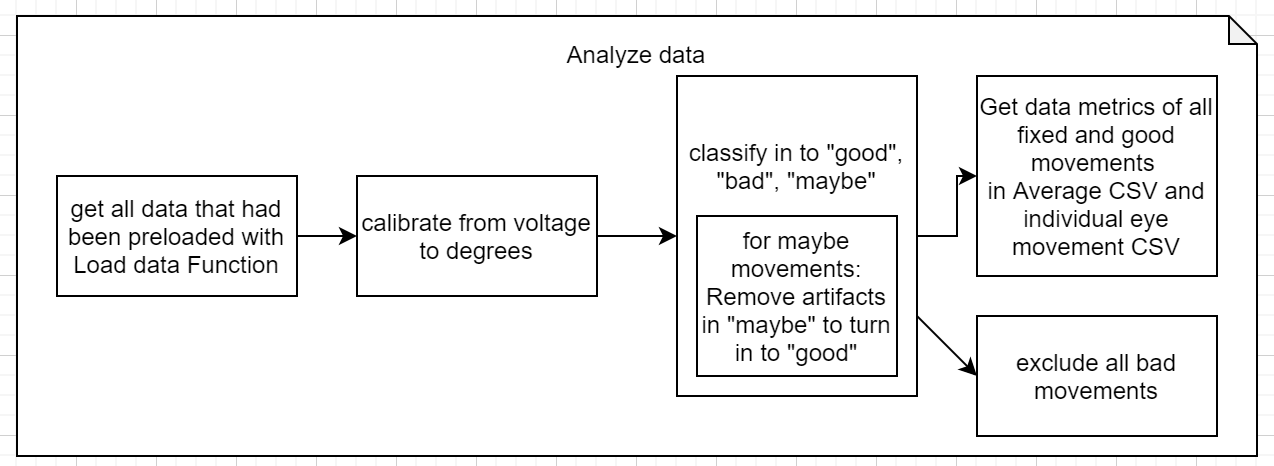
### [Test Case 040] Proper classification, artifact removal, and metrics for mixed data

### 

**Purpose:** To verify that mixed Raw Data can be selected and analyzed with RETINAS

**Specification:** REQ 50 -REQ 370

**Test Architecture:**



**Figure 1** *analyze flowchart*

**Equipment:**

1. Computer with RETINAS installed
2. Raw eye movement data file https://drive.google.com/open?id=1MjTsVpMIO7kVZfBj1xarF42vQ6iLP\_nQ
   1. Data file: Mixed eye data file
3. Internet connection

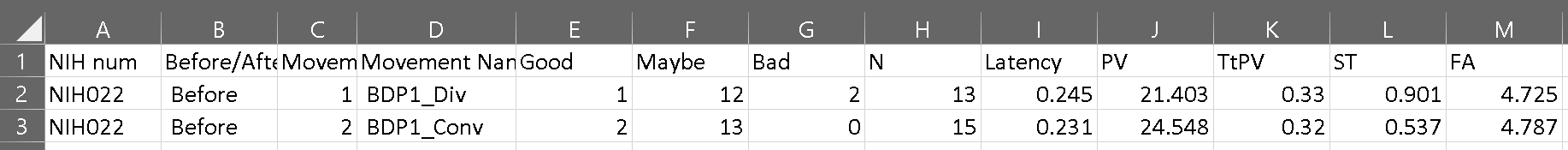
**Test Procedure**:

1. Prior to running retinas:
   1. Download the raw eye movement data files from the link in the equipment list item number 2
2. In Jupyter Notebook, Open RETINAS folder
3. Click to open Load data file
4. Click Run on the load data program
5. In the popup window, navigate to the “downloads” folder and select “rawData” folder that was previously downloaded
6. In the folder select the mixed data file and click OK.
7. Click to open the Analyze program
8. Click Run on the Analyze program
9. Once the program is done running and the output says “this took this many seconds:”, check the time that the program took, which will be the number that follows “seconds:”
10. To open the average result table
    1. Open Microsoft Excel
    2. Click file
    3. Click open
    4. Click browse
    5. Navigate to the “downloads” folder and select “avgoutputmetrics.txt”
    6. Click open
11. To open the individual result table
    1. Open Microsoft Excel
    2. Click file
    3. Click open
    4. Click browse
    5. Navigate to the “downloads” folder and select “fulloutputmetrics.txt”
    6. Click open
12. For reproducibility, click Run on the Analyze program again
13. Check if all previous measurements, timings, and classifications are identical

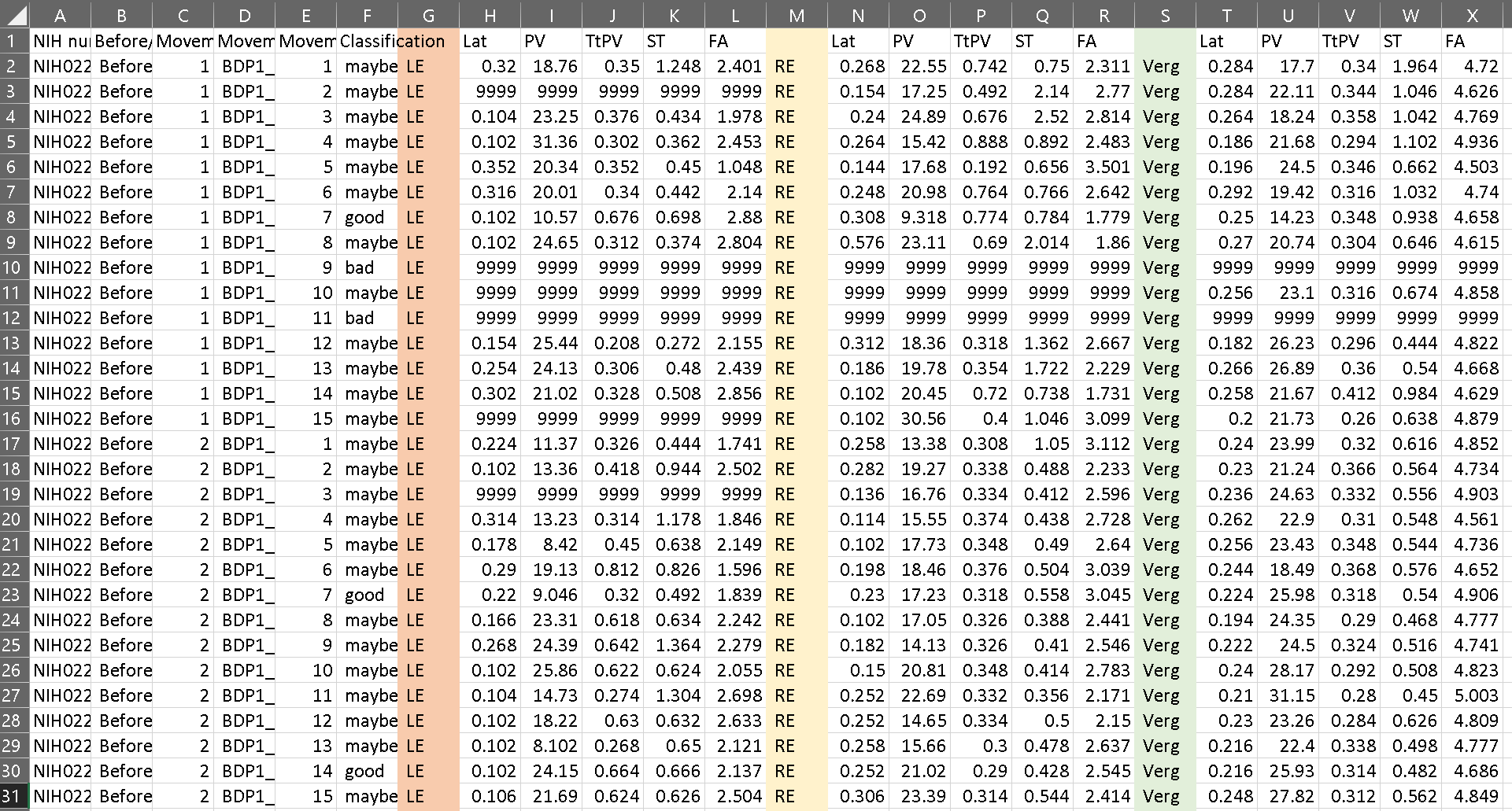
### 

Proper classification, artifact removal, and metrics for good data is done correctly if the following conditions are true:

1. The timing took under 45 seconds for the 30 files inserted via the test file

**Figure 2:**  Average Metrics outputted in CSV

1. Average Output CSV looks like Figure 2 (same number of columns and rows)
2. Average Output CSV has the same N values as Figure 2 (for good, maybe, bad, and total used)
3. Average Output CSV has the same 5 metrics as Figure 2 (with tolerance of 5%)



**Figure 3:**  Individual Metrics outputted in CSV

1. Individual output CSV looks like Figure 3 (same number of columns and rows)
2. Individual Output CSV has the same classification values as Figure 3 (for good, maybe, and bad)
3. Individual Output CSV has the same 5 metrics as Figure 3 (with tolerance of 5%)
4. Repeated run from step 12 has same outputs as previous run on both CSVs, including metrics, classifications, and timings on the terminal

***Test Results:***

|  |  |
| --- | --- |
| ***Item*** | ***Pass/Fail*** |
| Timing was less than 45 seconds | *Pass* |
| Average Output CSV looks like Figure 2 (same number of columns and rows) | *Pass* |
| Average Output CSV has the same N values as Figure 2 (for good, maybe, bad, and total used) | *Pass* |
| Average Output CSV has the same 5 metrics as Figure 2 (with tolerance of 5%) | *Pass* |
| Individual output CSV looks like Figure 3 (same number of columns and rows) | *Pass* |
| Individual Output CSV has the same classification values as Figure 3 (for good, maybe, and bad) | *Pass* |
| Individual Output CSV has the same 5 metrics as Figure 3 for each individual eye and the vergence movement (with tolerance of 5%) | *Pass* |
| Repeated run from step 12 has all same CSV outputs as previous run (with tolerance of 5%) | *Pass* |

### 

# 

# **9.** **ATTACHMENTS**

Cover Sheet for QUALITY RECORDS

|  |  |
| --- | --- |
| *Test Plan Name* | *RETINAS Test Plan* |
| *Test Engineer* | *RETINAS team* |
| *Test Date* | *5/1/20* |
| *Version* | *(HW)* ***(SW)*** |
| *System Name* |  |
| *Tests covered* | *Test 30-Test 40* |
| *MRs written?* | *Yes or* ***No*** *[see next page]* |
| *Data Attached?* | *Yes or* ***No*** |
| *Old Results appended?* | *Yes or* ***No*** |
| *Record Type* | *System Test Results* |
| *Date Filed* | *5/1/20* |
| *Storage Location (Room #)* |  |
| *Approval Signature(s)* |  |

|  |  |
| --- | --- |
| *Test Plan Name* | *RETINAS Test Plan* |
| *Test Engineer* | *RETINAS team* |
| *Test Date* | *3/1/20* |
| *Version* | *(HW)* ***(SW)*** |
| *System Name* |  |
| *Tests covered* | *Test 10-Test 20* |
| *MRs written?* | *Yes or* ***No*** *[see next page]* |
| *Data Attached?* | *Yes or* ***No*** |
| *Old Results appended?* | *Yes or* ***No*** |
| *Record Type* | *System Test Results* |
| *Date Filed* | *3/1/20* |
| *Storage Location (Room #)* |  |
| *Approval Signature(s)* |  |

***TEST STATUS***

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| ***Test Case*** | ***TEST PASSED?*** | ***IT TEST FAILED***  ***MR#*** | ***IF TEST FAILED, MR NOT ENTERED***  ***REASON*** | ***COMMENT*** |
| 010 | *Pass* |  |  | *Pass prior to restructuring the program to use offline CSV* |
| 020 | *Pass* |  |  | *Pass prior to restructuring the program to use offline CSV* |
| 030 | *Pass* |  |  |  |
| 040 | *Pass* |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |

***END OF DOCUMENT***