RuntimeTerror

Test plan

Version 2.2

04/16/2020

Document Control

Approval

The Guidance Team and the customer shall approve this document.

Document Change Control

|  |  |
| --- | --- |
| Initial Release: | 1.0 |
| Current Release: | 2.2 |
| Indicator of Last Page in Document: | $$$ |
| Date of Last Review: | 04/28/2020 |
| Date of Next Review: | 04/30/2020 |
| Target Date for Next Update: | 05/01/2020 |

Distribution List

This following list of people shall receive a copy of this document every time a new version of this document becomes available:

Guidance Team Members: Dr. Steven Roach

Jake Lasely

Customer: Mr.Baltazar Santella

Ms.Herandy Vasquez

Mr.Vincent Fonseca

Ms. Florencia Larsen

Mr. Eric de Nava

Software Team Members: Gilbert Alvarez

Jose Gallardo

Itzel Rivas

Eder Rodriguez

Nusrat Sarmin

Change Summary

The following table details changes made between versions of this document

|  |  |  |  |
| --- | --- | --- | --- |
| Version | Date | Modifier | Description |
| 1.0 | 04/10/2020 | Jose Gallardo | Created document |
| 1.1 | 04/10/2020 | Jose Gallardo | Added Purpose |
| 1.2 | 04/12/2020 | Itzel Rivas | Added Document Overview |
| 1.3 | 04/12/2020 | Eder Rodriguez | Added System Overview and Scope |
| 1.4 | 04/13/2020 | Gilbert Alvarez | Added Suspension and Exit Criteria |
| 1.5 | 04/14/2020 | Jose Gallardo | Added Section 2 |
| 1.6 | 04/16/2020 | Jose Gallardo  Eder Rodriguez  Itzel Rivas  Gilbert Alvarez | Added Section 3  Added Section 4, 4.1 (TEST XX)  Added Section 5 |
| 1.7 | 04/27/2020 | Jose Gallardo | Added to Test Approach, and Test Cases |
| 1.8 | 04/27/2020 | Itzel Rivas | Added to Test Approach and Test Cases, added to Test Features and Items and fixed Document Overview |
| 1.9 | 04/27/2020 | Eder Rodriguez | Added to Test Approach, Test cases and UI interface testing |
| 2.0 | 04/27/2020 | Gilbert Alvarez | UI interface testing |
| 2.1 | 04/28/2020 | Jose Gallardo | Added to Test Cases and some images to Appendix |
| 2.2 | 04/28/2020 | Jose Gallardo  Itzel Rivas  Gilbert Alvarez Eder Rodriguez | Reviewed Document and finalized Document |

Note: The template presented in this document was taken from:

Donaldson, S., and S. Siegel, *Successful Software Development*. Upper Saddle River, NJ: Prentice Hall, 2001, pp. 321-323.

Note: The template presented in this document was taken from: Donaldson, S., and S. Siegel, *Successful Software Development*. Upper Saddle River, NJ: Prentice Hall, 2001, pp. 321-323 and modified by Humberto Mendoza and Steve Roach.

Supplementary information is from:

Pfleeger, S. *Software Engineering, Theory and Practice*. Upper Saddle River, NJ: Prentice Hall, 1998, p. 365.

Table of Contents

[Document Control ii](#_Toc38998542)

[Approval ii](#_Toc38998543)

[Document Change Control ii](#_Toc38998544)

[Distribution List ii](#_Toc38998545)

[Change Summary ii](#_Toc38998546)

[1. Introduction 1](#_Toc38998547)

[1.1. Purpose 1](#_Toc38998548)

[1.2. Scope 1](#_Toc38998549)

[1.3. System Overview 1](#_Toc38998550)

[1.4. Suspension and Exit Criteria 1](#_Toc38998551)

[1.5. Document Overview 1](#_Toc38998552)

[1.6. References 2](#_Toc38998553)

[2. Test Items and Features 3](#_Toc38998554)

[3. Testing Approach 4](#_Toc38998555)

[4. Test XX 5](#_Toc38998556)

[3.1. Test PS TEST1 5](#_Toc38998557)

[3.2. Test PS TEST2 6](#_Toc38998558)

[3.3. Test PS TEST3 8](#_Toc38998559)

[3.4. Test GT TEST1 9](#_Toc38998560)

[3.5. Test GT TEST2 11](#_Toc38998561)

[3.6. Test PD TEST1 13](#_Toc38998562)

[3.7. Test PD TEST2 14](#_Toc38998563)

[4. User Interface Testing 15](#_Toc38998564)

[5. Test Schedule 18](#_Toc38998565)

[6. Other Sections 19](#_Toc38998566)

[7. Appendix 20](#_Toc38998567)

# Introduction

## Purpose

The purpose of a project test plan is to provide an overall structure to the development and test effort. This type of test plan will provide the overall strategy for testing the PMR Insight Collective Knowledge (PICK) system. Below you will find the components that will be tested, how they will be tested, and when they plan to be tested.

## Scope

This test plan will focus on the latest released version (v. 7.0) of the PICK System.

## System Overview

The system that will be tested is the PICK system and its interfacing with the third-party system Splunk.

## Suspension and Exit Criteria

Suspension of all testing will take place if fail rate of the total test cases that has occurred reaches 40%. This indicates that the number of defects prevents further testing on the system. Testing will terminate once there is a pass rate of 90%.

## Document Overview

The Document is divided into 7 main sections which are Introduction, Test Items and Features, Testing approach, Test, User Interface Testing, Test Schedule, Other Sections and Appendix. In the 1st section, Introduction, is divided into 6 subsections that will give an overview of the document. The 2nd section is Test Items and Features which describes the test items like the components, classes, functions between others in a table format. The 3rd section is Testing approach on which we will describe the approach that is being used by the system and specifying the types of tests performed. The 4th section, Test XX, is to add the test cases, in which it will document the test input, specific test procedures and outcomes. It will stablish the test methods and explain the nature and extent of each test. Within the 4th section there is User Interface Testing and focuses on the interaction between the user and the system. The 5th section, Test Schedule talks about the testing activities, here we will add a table with the order and completion dates of the tests. The 6th section, Other Components is in which we will add other sections that may appear in the test plan. The 7th section is the Appendix will contain the extra information we used in the document.

## References

[1] SDD Team 7

# Test Items and Features

|  |  |
| --- | --- |
| Features | Being Tested |
| Splunk | Connecting to Splunk |
| Ingestion | Uploading log files to Splunk |
| Ingestion | Parsing results of Splunk to Log Entry objects |
| Cleansing | Correct lines removed |
| Graph | Adding of nodes |
| Graph | Adding of edges |
| Database | Storing vectors and log entries into their collections |
| Database | Returning from vector and log entries collection |

# Testing Approach

Table 1: Test Plan Splunk

|  |  |  |
| --- | --- | --- |
| **TEST SUITE <Splunk>** | | |
| **Description of Test Suite** | **This test suit will focus on testing the interaction of the PICK system and the Splunk system** | |
| **Test Case Identifier** | **Objective** | **Criticality** |
| PS TEST1 | Connect to Splunk | Critical |
| PS TEST2 | Ingest Log Files | Critical |
| PS TEST3 | Return Indexed Log Entries | Critical |

Table 2: Test Plan Graph

|  |  |  |
| --- | --- | --- |
| **TEST SUITE <Graph>** | | |
| **Description of Test Suite** | **This test suit will focus on testing the Graphing components of the PICK system** | |
| **Test Case Identifier** | **Objective** | **Criticality** |
| GT TEST1 | Add a Node | Critical |
| GT TEST2 | Add an Edge | Critical |

Table 3: Test Plan Database

|  |  |  |
| --- | --- | --- |
| **TEST SUITE <Database>** | | |
| **Description of Test Suite** | **This test suit will focus on testing the Database components of the PICK system** | |
| **Test Case Identifier** | **Objective** | **Criticality** |
| PD TEST1 | Connect to Database | Critical |
| PD TEST2 | Store Data into Vector Collection | Critical |
| PD TEST3 | Store Data into Log Entry Collection | Critical |
| PD TEST4 | Return Data from Vector Collection | Critical |
| PDTEST 5 | Return Data from Log Entry Collection | Critical |

Table 4: Test Plan Cleansing

|  |  |  |
| --- | --- | --- |
| **TEST SUITE <Cleansing>** | | |
| **Description of Test Suite** | **This test suit will focus on testing the interaction of the PICK system and the Log Files in the root directory** | |
| **Test Case Identifier** | **Objective** | **Criticality** |
| PC TEST1 | Cleanse Log Files | Critical |

# Test XX

## Test PS TEST1

**Objective:**  To test the process of connecting Splunk to the system

**Notes:**

A Splunk config file exist that the user has to place their credentials into the appropriate fields to connect to Splunk.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Test No.: PS TEST 1 | | | | Current Status: Passed | | |
| Test title: Connect to Splunk | | | | | | |
| Testing approach: This test will be conducted using the 8.0 version of Splunk installed on a Linux machine. Behavior of the is observed on the monitor of the machine being used. | | | | | | |
| STEP  1 | OPERATOR ACTION  Provide Splunk Admin credentials in Splunk Config file. | PURPOSE  Initial Condition | | | EXEPCTED RESULTS  User information is stored and ready to be used when called upon | COMMENTS |
| STEP  2 | Click on file new project | Begins project creation | | | Project Configurations are displayed |  |
| STEP  3 | In Directory Configuration click browse and select root directory | Stores the information for root directory | | | Root directory is saved and displayed in Directory configuration view |  |
| STEP  4 | Press button “Start new project” | Begins connection to Splunk | | | Splunk connection successful |  |
| Concluding Remarks: | | | | | | |
| Testing Team:  Lead: Eder Rodriguez  Jose Gallardo  Itzel Rivas  Gilbert Alvarez | | | Date Completed:  4/15/2020 | | | |

## Test PS TEST2

**Objective:** To test the process of connecting Splunk to the system and ingesting the files to Splunk

**Notes:** A Splunk config file exist that the user has to place their credentials into the appropriate fields to connect to Splunk.

Pre-Condition: Root Directory has been established and has log files to be ingested

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Test No.: PS TEST 2 | | | | Current Status: Passed | | |
| Test title: Ingest Log Files | | | | | | |
| Testing approach:  This test will be conducted using the 8.0 version of Splunk installed on a Linux machine. Behavior of the is observed on the monitor of the machine being used. | | | | | | |
| STEP  1 | OPERATOR ACTION  New Project has started | PURPOSE  Initial Condition | | | EXEPCTED RESULTS  Project information is stored and connection with Splunk is created | COMMENTS |
| STEP  2 | View Status of Log Files | Displays whether a log file has been ingested | | | Green Checkmark next to file that has been ingested |  |
| Concluding Remarks: | | | | | | |
| Testing Team:  Lead: Eder Rodriguez  Jose Gallardo  Itzel Rivas  Gilbert Alvarez | | | Date Completed:  4/21/2020 | | | |

## Test PS TEST3

**Objective:** To test the results of ingestion

**Notes:**

A Splunk config file exist that the user must place their credentials into the appropriate fields to connect to Splunk.

Pre-Condition: Root Directory has been established and has log files to be ingested

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Test No.: PS TEST 3 | | | | Current Status: Passed | | |
| Test title: Return Indexed Log Entries | | | | | | |
| Testing approach: This test will be conducted using the 8.0 version of Splunk installed on a Linux machine. Behavior of the is observed on the monitor of the machine being used. | | | | | | |
| STEP  1 | OPERATOR ACTION  From Log Processing View Switch to Analysis View | PURPOSE  Initial Condition | | | EXEPCTED RESULTS  Analysis View is displayed | COMMENTS |
| STEP  2 | User views returned entries from Splunk | Allows user to correlate entries | | | Log Entries are displayed in analysis view |  |
| Concluding Remarks:  Log Entries are displayed barring interruption from splunk connection | | | | | | |
| Testing Team:  Lead: Eder Rodriguez  Jose Gallardo  Itzel Rivas  Gilbert Alvarez | | | Date Completed:  4/21/2020 | | | |

## Test GT TEST1

**Objective:** To test the process of adding a node to the graph

**Notes:**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Test No.: GT Test 1 | | | | Current Status: Passed | | |
| Test title: Add Node | | | | | | |
| Testing approach: This test will be conducted using the .29 version of QGraphViz | | | | | | |
| STEP  1 | OPERATOR ACTION  From Analysis View, switch to Vector View | PURPOSE  Initial Condition | | | EXEPCTED RESULTS  Vector View is displayed with empty canvas | COMMENTS |
| STEP  2 | User clicks on add node button | Begins steps of node creation | | | Node Creation Pop up displayed |  |
| STEP  3 | User enters  “Database” for Node name  “SQL” for label  And selects “box” for type and presses OK | Node is created | | | Node is displayed in Graph canvas |  |
| Concluding Remarks:  Node can be easily created, but some can have the same name. | | | | | | |
| Testing Team:  Lead: Eder Rodriguez  Jose Gallardo  Itzel Rivas  Gilbert Alvarez | | | Date Completed:  4/24/2020 | | | |

## Test GT TEST2

**Objective:** To test the process of adding a relationship between nodes

**Notes:**

Pre-Condition: At least 2 Node must be created.For Test Purposes add one node with the following Name: “MongoDB”

Label: “MongoDB”

Type: box

And another Node with the following

Name: “Server”

Label: “Server”

Type: “Circle

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Test No.: GT Test 2 | | | | Current Status: Passed | | |
| Test title: Add Edge | | | | | | |
| Testing approach: This test will be conducted using the .29 version of QGraphViz | | | | | | |
| STEP | OPERATOR ACTION  From Analysis View, switch to Vector View | PURPOSE  Initial Condition | | | EXEPCTED RESULTS  Vector View is displayed with empty canvas | COMMENTS |
| STEP  1 | User Clicks on add edge button | Initial Condition | | | Edge pop-up is displayed |  |
| STEP  2 | User selects source node as MongoDB and destination node as Server then presses ok | Defines the connection between nodes | | | Relation is displayed |  |
| Concluding Remarks:  The functionality of the graph worked, however the destination node begins with the same source node. | | | | | | |
| Testing Team:  Lead: Jose Galardo  Eder Rodriguez  Itzel Rivas  Gilbert Alvarez | | | Date Completed:  4/24/2020 | | | |

## Test PD TEST1

**Objective:**  To test the process of connecting MongoDB Database to the system

**Notes:**

Download MongoDB community version 4.2.6

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Test No.: PD TEST 1 | | | | Current Status: Passed | | |
| Test title: Connect to MongoDB Database | | | | | | |
| Testing approach:  The test will be conducted using the 4.2.6 version of MongoDB | | | | | | |
| STEP  1 | OPERATOR ACTION  Open MongoDB App on desktop | PURPOSE  Initial Condition | | | EXEPCTED RESULTS  Prompt to add a new connection | COMMENTS |
| STEP  2 | Add localhost string (I.e. mongodb://localhost:27017/) | Stablish a connection to the localhost | | | Connection to local host successful |  |
| Concluding Remarks:  The connection string is connected automatically to the program with MongoClient. | | | | | | |
| Testing Team:  Lead: Eder Rodriguez  Jose Gallardo  Itzel Rivas  Gilbert Alvarez | | | Date Completed:  04/26/2020 | | | |

## Test PD TEST2

**Objective:**  To test the process of connecting Splunk to the system

**Notes:**

There are 3 folders called Red, White and Blue that contain log files that the user has to place in the root directory to start ingestion.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Test No.: PD TEST 2 | | | | Current Status: Pending | | |
| Test title: Store to Vector Collection | | | | | | |
| Testing approach:  The test will be conducted using the 4.2.6 version of MongoDB and PyMongo  The program automatically connects to the Database by having the connection string in the client. It needs to be connected first to the app on desktop.  Pre-Condition:  MongoDB is connected to the localhost | | | | | | |
| STEP  1 | OPERATOR ACTION  Click on "New Project” on the upper left corner of the system. | PURPOSE  Initial Condition | | | EXEPCTED RESULTS  New project is created | COMMENTS |
| STEP  2 | Switch to “Vector Configuration” click  “Add Vector” | Display all the properties of the vector to be added | | | Add all the properties of the vector. (i.e. name, description, etc.) |  |
| STEP 3 | Click “Submit” to Add vector to the database collection | Add vector | | | Vector added into the Database. |  |
| Concluding Remarks: | | | | | | |
| Testing Team:  Lead: Gilbert Alvarez  Jose Gallardo  Itzel Rivas | | | Date Completed:  TBA | | | |

## Test PD TEST3

**Objective:**  To test the process of adding a log entry into a database collection.

**Notes:**

There are 3 folders called Red, White and Blue that contain log files that the user has to place in the root directory to start ingestion.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Test No.: PD TEST 3 | | | | Current Status: Pending | | |
| Test title: Store to Log Entry Collection | | | | | | |
| Testing approach:  The test will be conducted using the 4.2.6 version of MongoDB and PyMongo  The program automatically connects to the Database by having the connection string in the client. It needs to be connected first to the app on desktop.  Pre-condition:  MongoDB is connected to the localhost  A new project must be created. | | | | | | |
| STEP  1 | OPERATOR ACTION  Under “Directory Configuration” add root directory | PURPOSE  Initial Condition | | | EXEPCTED RESULTS  Log files are selected | COMMENTS |
| STEP 2 | Click “Save” under Root Directory | Log files will be saved into the system | | | Log files are added to the system |  |
| STEP  2 | Switch to “Log Processing view” to view the files that are cleansed and validated. | Log files are prepared to be ingested | | | Log files are ready for ingestion |  |
| STEP 3 | The log files that are ingested convert into log entries. | Log entries need to be stored into the database | | | Log entries are stored into the database collection. |  |
| Concluding Remarks: | | | | | | |
| Testing Team:  Lead: Eder Rodriguez  Gilbert Alvarez | | | Date Completed:  TBA | | | |

## Test PD TEST4

**Objective:**  To test the process of returning a vector from the database collection.

**Notes:**

Pre-condition:

MongoDB is connected to the localhost. Also, connection to Splunk and ingestion has occurred.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Test No.: PD TEST 4 | | | | Current Status: Pending | | |
| Test title: Return Vector from Collection | | | | | | |
| Testing approach:  The test will be conducted using the 4.2.6 version of MongoDB and PyMongo | | | | | | |
| STEP  1 | OPERATOR ACTION  Click on “Analysis View” tab | PURPOSE  Initial Condition | | | EXEPCTED RESULTS  Analysis view is displayed. | COMMENTS |
| STEP  2 | Click on “Vector View” tab. | Vectors are being displayed | | | Successfully return all the vectors stored in the database collection |  |
| Concluding Remarks: | | | | | | |
| Testing Team:  Lead: Eder Rodriguez  Jose Gallardo  Itzel Rivas  Gilbert Alvarez | | | Date Completed:  TBA | | | |

## Test PD TEST5

**Objective:**  To test the process of adding a log entry into a database collection.

**Notes:**

Pre-condition:

MongoDB is connected to the localhost. Also, connection to Splunk and ingestion has occurred.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Test No.: PD TEST5 | | | | Current Status: Pending | | |
| Test title: Return Log Entry from database Collection | | | | | | |
| Testing approach:  The test will be conducted using the 4.2.6 version of MongoDB and PyMongo | | | | | | |
| STEP  1 | OPERATOR ACTION  Click on “Analysis View” tab | PURPOSE  Initial Condition | | | EXEPCTED RESULTS  Analysis view is displayed. | COMMENTS |
| STEP  2 | Click on “Log Entries” tab | Log entries need to be displayed | | | Log Entries are successfully returned from database collection |  |
| Concluding Remarks: | | | | | | |
| Testing Team:  Lead: Itzel Rivas  Eder Rodriguez  Gilbert Alvarez | | | Date Completed:  TBA | | | |

## Test PC TEST1

**Objective:** To test the cleansing process of the system

**Notes:**

Pre-Condition: Root Directory has been established and has log files to be cleansed

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Test No.: PC TEST 1 | | | | Current Status: Pending | | |
| Test title: Cleanse Log Files | | | | | | |
| Testing approach:  This test will be focusing on the removal of empty lines from log files | | | | | | |
| STEP  1 | OPERATOR ACTION  Click File and New project | PURPOSE  Initial Condition | | | EXEPCTED RESULTS  Project Configuration is displayed | COMMENTS |
| STEP  2 | In Directory configuration click browse and select “root” from the test folder. | Stores root directory for files | | | Root directory path is displayed in view |  |
| STEP  3 | Click on start new project | Begins the cleansing process | | | Log Processing View is displayed |  |
| STEP  4 | View cleansing status of log files | Displays whether a log file has been cleansed | | | Green Checkmark next to file that has been cleansed |  |
| Concluding Remarks: | | | | | | |
| Testing Team:  Lead: Jose Gallardo  Gilbert Alvarez | | | Date Completed:  TBD | | | |

# User Interface Testing

**(References screenshots found in Appendix under “Section 4”)**

1) Creation of a new project starts by clicking the file item on the menu bar (*UI figure 1.0*).

2) ­New project Opens up a dialog box that will allow the user to enter information needed for

a) Team Configuration: There is a label at the top of the view to notify the user they are on the dialog to enter data for the Team Configuration. (*UI figure 1.2*)

1. Radial button for a user to select if they are the lead analyst.
2. Input field with a label marked Lead IP address for the lead analyst to enter their IP address.
3. Field listing all machines connected to the host.
4. For analysts that are not the lead, input for the Lead IP address is needed for them to connect to the host machine.
5. A “Connect” button allows users to connect to the hosts machine and assures user that they have connected to the hosts machine.

b) Event Configuration: There is a label at the top of the view to notify the user they are on the dialog to enter data for the Event Configuration. (*UI figure 1.1*)

1. Input fields with appropriate labels next to them to notify the user the user the type of data they should input. This includes the Event name, Event Description, Start date/time and End date/time/
2. Save button to reassure the data input is saved.

c) Directory Configuration: There is a label at the top of the view to notify the user they are on the dialog to enter data for the Team Configuration. (*UI figure 1.4*)

1. Labels of the target directory are placed next to the input fields that allows the user to type in a directory path.
2. Browse buttons next to each field allow the user to search for the directory with the file explorer if the user cannot remember the full directory path.

d) Vector Configuration: There is a label at the top of the view to notify the user they are on the dialog to enter data for the Team Configuration. (*UI figure 1.3*)

1. A field for the user to view the vectors they have created. They may choose by clicking on a vector row and hitting the button labeled “Delete Vector” at the bottom.
2. Buttons at the bottom of the view are labeled appropriately for the user to choose based on the action they want to take. These buttons include “Delete Vector”, “Edit Vector” and “Add Vector”.

* Add Vector changes the view to allow the user to create a new vector with a field for the Vector Name, Vector Description and a “Submit” button a the bottom to complete the process of creating a vector. (*UI figure 1.5*)

At the bottom of the dialog box is a button labeled “Start New Project” that allows the user to continue onto the next process after they have finished inputting all the necessary data, and a cancel button that lets allows the user to abort the whole process of inputting data.

From the Start Project Button, the user is taken to a new view that contains a table view rows and tabs to change views to the analysis view. The log entry view (UI figure 2.1) contains the table view that will display information on the log entries such as their source log file, cleansing, validation, ingestion status and a selection row. The button at the bottom of this view, allows the user to start the analysis process.

There are no grammatical errors or misspelled words in this portion of the User Interface. Each Input Field and button are labeled properly, and each view presents the appropriate UI element for the user to enter the necessary data to start a new project.

The analysis view (UI figure 2.0) shows the graph (UI figure 2.3) and log entries table view. At any point the user can edit vector configuration by accessing the “Edit Vector Configuration” menu item in the Edit menu (UI figure 2.2). The UI for the log entry view and analysis view doesn’t contain grammatical errors nor misspellings, it is missing some labels like the filter and search that might confuse the user and not allow them to find these features.

In Progress:

-There is no current process of alerting the user if there is data missing or they need to input more information. (UI figure 1.1 – UI figure 1.5)

- There are no labels to help the user identify the search and filter bar. (UI figure 2.0 and UI figure 2.1)

# Test Schedule

|  |  |  |
| --- | --- | --- |
| **Task and date** | **People** | **Description** |
| Splunk: 04/15 | Eder Rodriguez, Itzel Rivas, Jose Gallardo, Gilbert Alvarez | Testing the connection |
| Ingestion: 04/18 | Eder Rodriguez, Gilbert Alvarez | Testing uploading of files to Splunk and parsing the results into Log Entries. |
| Cleansing: 04/18 | Jose Gallardo, Itzel Rivas | Testing the cleansing process to remove unwanted lines in the log file |
| Graph: 04/20 | Eder Rodriguez, Jose Gallardo, Itzel Rivas | Testing the creation of nodes. |
| Graph: 04/20 | Eder Rodriguez, Jose Gallardo, Gilbert Alvarez | Testing the creation of edges |
| Database: 04/26 | Itzel Rivas, Jose Gallardo, Eder Rodriguez | Testing the connection of the database |

# Other Sections

Environmental Requirements: a machine preferably running on Kali Linux and hard drive with enough storage for storing the root directory of the log file hierarchy.

Software Requirements: Python 3, PyQt 5 library, QGraph Viz, Splunk, Splunk SDK, Splunk lib, PyMongo and Mongo DB.

# Appendix

**Section 4**

A close up of a sign

Description automatically generated

*UI figure 1.0*

A screenshot of a cell phone

Description automatically generated

*UI figure 1.1*

A screenshot of a cell phone

Description automatically generated

*UI figure 1.2*

A screenshot of a cell phone

Description automatically generated

*UI figure 1.3*

A screenshot of a cell phone

Description automatically generated

*UI figure 1.4*

A screenshot of a social media post

Description automatically generated

*UI figure 1.5*

A screenshot of a social media post

Description automatically generated

*UI figure 2.0*

A screenshot of a social media post

Description automatically generated

*UI figure 2.1*

A screenshot of a cell phone

Description automatically generated

*UI figure 2.2*

A screenshot of a social media post

Description automatically generated

*UI figure 2.3*

*$$$*