
<Enter team name here>

**Prevent, Mitigate, and Recover (PMR) Insight
Collective Knowledge System (PICK)
Software Requirements Specification**
Version 0.1
2/5/2020

Table of Contents

<u>1. SOFTWARE CONFIGURATION MANAGEMENT</u>	
<u>1.1. SOFTWARE CONFIGURATION IDENTIFICATION</u>	3
<u>1.2. SOFTWARE CONFIGURATION CONTROL</u>	3
<u>1.3. SOFTWARE CONFIGURATION AUDITING</u>	3
<u>1.3.1. Configuration Items</u>	3
<u>1.3.2. Document Change Control</u>	4
<u>1.4. Software Configuration accounting</u>	4
<u>1.5. REFERENCES</u>	4

1. Software Configuration Management

1.1. Software Configuration Identification

1.1.1 Configuration Items:

Configuration Item	Description
ConfigGraphFormat.py	The nodes configuration in graphical format window allows the user to display and search nodes in a graphical representation. The user can interact with a timeline, select interval units and zoom in/out on the graph.
ConfigTabFormat.py	The nodes configuration in tabular format window provides all the nodes along their metadata in a table. The user can interact with the table and edit the properties of each node.
FilterConfig.py	The filter configuration interface displays a variety of different criteria that the user can specify to filter logs in the system. These criteria include a keyword search, author of the log, event type, start timestamp and end timestamp.
GraphConfiguration.py	The graph configuration interface displays a description of the vector that will be displayed as a graph. The user can change the vector through a dropdown menu. In the graph section of the interface, the user can add, delete, or edit both a relationship and a node.
IconConfiguration.py	The icon configuration interface displays a table that allows the user to select a given icon. Name, source, and image of such icon is displayed in the table. The user can choose to add, delete, or edit the icon.
LogFileConfig.py	The log file configuration window gives a table view of all log files, their source, cleansing status, validation status and ingestion status. This window allows the user to view the enforcement action report for any log file, which lists any errors found in a log file. From the enforcement action report, the user can manually validate any file.
PICK_PMR.py	This is a window that allows the user to access all interface pages of the project by using tabs.
RelationshipsConfig.py	The relationship configuration window displays the relationships among vectors in a table, and allows the user to add, remove or edit any relationship that is found on the table.
VectorConfig.py	This interface displays a table view of vectors and vector descriptions. Vectors can be added, deleted, or edited from this interface.
VectorDBConfigurationLead.py	This interface allows the lead of the project to approve and commit any given vector from the vector database.

VectorDBConfigurationNon-Lead.py	Allows non-lead user to choose a vector from the database to pull a version or push a version for approval from the lead. The connection status to the lead is displayed in the top portion of the interface.
change_configuration.py	The change configuration will display a list of changes made to a log entry.
directoryConfiguration.py	Directory Configuration's main window. Allows the analyst to provide a root directory, along with a red, blue and white team folder paths.
eventConfiguration.py	Event Configuration window. Allows the analyst to provide an event name, event description, event start timestamp and event end timestamp
export_configuration.py	The export configuration is responsible for controlling the export format of a graph.
log_entry_configuration.py	The log entry configuration displays information about entries in the system. An analyst can view metadata about a log entry such as the source, host, source type and contents of the log entry as well as assign the entry to a vector.
teamConfiguration.py	Team configuration main window. It allows the analyst to provide a Lead IP address, along with the number of established connections to the lead's IP address.

1.1.2 Document Change Control

Initial Release:	0.1
Current Release:	0.1
Indicator of Last Page in Document:	&
Date of Last Review:	02/05/2020
Date of Next Review:	02/07/2020
Target Date for Next Update:	02/18/2020

1.2. Software Configuration Control

New versions of the configuration are created from the feedback received from the clients. Changes made/features added will be made based on client feedback/requirements or if a more optimal solution is found that is within the scope of the client's needs. The priority of the project is always to deliver the clients' needs. Responsibilities are assigned by taking a poll with respect to an individuals' interest on an area in the project. The source code control tool used in our configuration is Git VCS. For error documentation we will use the Zoho Bugtracker software. This software will give us the ability to describe the errors we face, as well as who will oversee fixing them. Our directory structure will follow a hierarchical architecture: the GUI components will go on one package, icons on another one, and business logic on a separate package.

1.3. Software Configuration Auditing

The Guidance Team and the customers shall approve the degree to which the configuration matches the desired configuration. (Gates, 2020)

1.4. Software Configuration Status Accounting

Mechanism for determining (and recording) a record of where a version of the system is at any point in time with respect to what the version in its final state should be

1.5. References

Dr. Gates, et al. *Prevent, Mitigate, and Recover (PMR) Insight Collective Knowledge System (PICK) Software Requirements Specification* . 2020, pp. 1–27, *Prevent, Mitigate, and Recover (PMR) Insight Collective Knowledge System (PICK) Software Requirements Specification* .

H. E. Bersoff, *Elements of Software Configuration Management*. *IEEE Transactions on Software Engineering*, 10(1):79-87, Jan 1984

Hans Van Vliet, *Software Engineering, Principles and Practice*, 3rd edition, John Wiley & Sons, 2008. Chapter4.

&