

**PMR Insight Collective Knowledge (PICK)  
Class-responsibility-collaboration Model**

**Version 1.0  
2/18/2020**

# Document Control

## Approval

The Guidance Team and the customer shall approve this document.

## Document Change Control

Initial Release:	1.0
Current Release:	1.0
Indicator of Last Page in Document:	[END]
Date of Last Review:	02/18/2020
Date of Next Review:	02/18/2020
Target Date for Next Update:	02/18/2020

## 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. Ann Gates, Dr. Steve Roach, Mr. Jake Lasley

Clients:

Mr. Anthony DesArmier, Mr. Angel Villalpando, Mr. Jorge Garcia, Mr. David Rayner, Mr. Valentin Becerra, Mr. Mario Delgado

Software Team Members:

Mr. Anthony DesArmier, Mr. Angel Villalpando, Mr. Jorge Garcia, Mr. David Rayner, Mr. Valentin Becerra, Mr. Mario Delgado

## Change Summary

The following table details changes made between versions of this document

Version	Date	Modifier	Description
0.1	2/14/2020	David Rayner	Added template Added Settings, Node, Graph, ProjectMerge, History, UIChangeConfig, UIVectorConfig, UI
0.1	2/14/2020	Angel Villalpando	Added UiDirectoryConfig, UiEventConfig, UiFilterConfig, UiVectorDBLead, LogFile, EnforcementActionReport, Validator, ExportTable, ExportGraph
0.2	2/16/2020	Jorge Garcia	Added UiVectorConfig, formatting
0.2	2/16/2020	Valentin Becerra	Node, Log Entry, Graph, Vector
0.3	2/17/2020	Mario Delgado	Edited document, made changes to UIVectorConfig
1.0	2/17/2020	Anthony DesArmier	Verified UiDirectoryConfig, UiEventConfig, UiFilterConfig, UiVectorDBLead, LogFile, EnforcementActionReport, Validator, ExportTable, ExportGraph

CRC	Keikaku 企画	Date 2/18/2020 2:39 AM	Page 1
-----	------------	---------------------------	-----------

**Class:** UiChangeConfig

**Description:** Commit window handles committing of log entry and vector changes.

**Responsibilities:**

1. Display vector changes
2. Display node changes.
3. Display relationship changes.
4. Display log entry changes.
5. Commit button triggers save action.

**Collaborators:**

- History

**Class:** UiDirectoryConfig

**Description:** Handles the setting of file or file path directories.

**Responsibilities:**

1. Collects user provided directory path for root ingestion.
2. Collects red team directory path.
3. Collects blue team directory path.
4. Collects white team directory path.

**Collaborators:**

- Settings

**Class:** UiEventConfig

**Description:** Handles the setting of project event information.

**Responsibilities:**

1. Collects event name.
2. Collects event description.
3. Collects event start timestamp.
4. Collects event end timestamp.

**Collaborators:**

- Event

**Class:** UiExportConfig

**Description:** The export window which handles exporting vector and entry data to an external file.

**Responsibilities:**

1. Allows user to select directory.
2. Allows user to select file format.
3. Allows user to name file to export.

**Collaborators:**

- TableExport
- GraphExport

**Class:** UiFilterConfig

**Description:** Collects, handles, and applies the search criteria to the system generated log entries.

**Responsibilities:**

1. Collects keywords to search for in generated log entries.
2. Collects the start/end timestamp search-criteria to apply to generated log entries.
3. Collects the “creator” (red, white, or blue) search criteria for generated log entries.
4. Collects the “event type” (red, white, or blue) search-criteria for generated log files.

**Collaborators:**

- Settings

CRC	Keikaku 企画	Date 2/18/2020 2:39 AM	Page 2
-----	------------	---------------------------	-----------

**Class:** UiTeamConfig

**Description:** The window which handles the system role and connections to the host when running as a client.

**Responsibilities:**

1. Allows user to define if they are a lead or not.
2. If lead is checked, displays the lead's IP address.
3. If lead is not checked, displays the IP address of the lead's network adapter.
4. Displays the number of connections to the lead's IP address.

**Collaborators:**

- Settings

**Class:** UiVectorConfig

**Description:** The vector window which handles the adding, editing, and deleting of vectors and their descriptions.

**Responsibilities:**

1. Allows user to add a new vector name.
2. Allows user to add a new vector description.
3. Allows user to delete a vector.
4. Allows user to edit existing vector attributes.

**Collaborators:**

- Vector.

**Class:** UiRelationshipConfig

**Description:** Handles the relationship table for the active vector.

**Responsibilities:**

1. Display relationship id.
2. Collects parent id.
3. Collects child id.
4. Allows user to enter label description.
5. Allows user to add new relationship entry.
6. Allows user to remove a relationship entry.
7. Allows user to edit relationship entry.

**Collaborators:**

- Relationship

**Class:** UiVectorDBAnalyst

**Description:** The analyst vector DB window which handles the pulling and pushing of vector changes between the analyst and host systems.

**Responsibilities:**

1. Display vector lead's pull table.
2. Display vector Analyst push table.
3. Allows user to pull lead's table.
4. Allows user to push Analyst table.

**Collaborators:**

- Settings

CRC	Keikaku 企画	Date 2/18/2020 2:39 AM	Page 3
-----	------------	---------------------------	-----------

**Class:** UiVectorDBLead

**Description:** The lead vector DB window which handles the accepting of queued analyst push requests.

**Responsibilities:**

1. Allows user to accept push requests to the database.
2. Allows user to decline push requests to the database.
3. Display information to be pushed to the database.

**Collaborators:**

- ProjectMerge

**Class:** UI

**Description:** The main window which serves as entry point to the application and provides the bulk of the system's interface.

**Responsibilities:**

1. Launching all UI dialogs.
2. Display vector table.
3. Allows user to add, edit and delete nodes.
4. Allows user to select node property visibilities.
5. Allows user to set node visibility.
6. Display vector graphical view.
7. Allows user to zoom in/out on the graphical view.
8. Allows user to change the timeline orientation
9. Allows user to change the timeline interval
10. Allows user to select between different vectors.
11. Displays log entry table
12. Allows user the ability to edit log entries.
13. Displays log file able
14. Displays Enforcement Action Report table.
15. Allows user to select an Enforcement Action Report to display.
16. Allows user to revalidate log files.
17. Allows user to filter log file, log entry, and node table.

**Collaborators:**

- Node
- Vector
- Graph
- UIFilterConfig
- UiVectorDBLead
- UiVectorDBAnalyst
- UiRelationshipConfig
- UiVectorConfig
- UiTeamConfig
- UiExportConfig
- UiDirectoryConfig
- UiChangeConfig

**Class:** History

**Description:** Keeps track of the log entries, nodes, and vectors and relationships history of changes.

**Responsibilities:**

1. Allows user to undo commit action.
2. Stores data in sequence.
3. Provides commit information.

**Collaborators:**

- Node
- LogEntry
- Relationship
- Vector

**Class:** Event

**Description:** Maintains the information and attributes that describe the specific event in a given analysis/session.

**Responsibilities:**

1. Stores the user-provided event name.
2. Stores the user-provided event description.
3. Stores the user-provided start timestamp.
4. Stores the user-provided end timestamp.
5. Stores the vector names for an event.

**Collaborators:**

- UiEventConfig

**Class:** LogFile

**Description:** The log file class will read the user-provided directory path and import the contents of the file.

**Responsibilities:**

1. Reads the directory path provided by the user.
2. Cleanses with help of validator class.
3. Validates with help of validator class.
4. Keeps track of status of current cleansed or validated states.
5. Creates copy of cleansed files.
6. Sends files to Splunk for processing.

**Collaborators:**

- Validator
- EnforcementActionReport

**Class:** Graph

**Description:** Keeps track of nodes, relationships, their relative positions, and their properties.

**Responsibilities:**

1. Stores node positions.
2. Stores relationship positions.
3. Stores graph orientation.
4. Store timeline intervals.

**Collaborators:**

- Node
- Relationship

CRC	Keikaku 企画	Date 2/18/2020 2:39 AM	Page 5
-----	------------	---------------------------	-----------

**Class:** Node

**Description:** Responsible for the storage of log entry properties.

**Responsibilities:**

1. Stores information about relevant Log Entry (if applicable).
2. Stores reference to icon image path that Graph will use to render Node.
3. Generates a unique node id.
4. Keep track of node visibility.
5. Keeps track of node coordinate position.

**Collaborators:**

- Settings

**Class:** Vector

**Description:** Responsible for storing the data of nodes and relationships that compose a vector.

**Responsibilities:**

1. Stores node IDs of nodes in a given vector.
2. Stores relationship IDs between the nodes in a Vector.
3. Keeps track of log entry property visibility flags.

**Collaborators:**

- Node
- Relationship

**Class:** Relationship

**Description:** Stores the child and parent node IDs that constitute a given relationship in the system.

**Responsibilities:**

1. Stores the node identification number for a given parent node.
2. Stores the node identification number for a given child node.
3. Generates a relationship ID for a parent and a child node.

**Collaborators:**

- Node

**Class:** EnforcementActionReport

**Description:** Report that outlines various errors found in validation and solicits the user for an enforcement action.

**Responsibilities:**

1. Collects the errors detected by Validator.
2. Determines what action must be performed depending on the error detected.

**Collaborators:**

- Validator

**Class:** NewProject

**Description:** Handles the transition between Events and the saving of such events.

**Responsibilities:**

1. Saves Events.
2. Loads Events.
3. Manages changes between Events.

**Collaborators:**

- Event

CRC	Keikaku 企画	Date 2/18/2020 2:39 AM	Page 6
-----	------------	---------------------------	-----------

**Class:** ProjectMerge

**Description:** Handles the merging of analyst and lead changes.

1. Manages merge of data for analyst on pull of lead changes.
2. Manages merge of data for lead on push of analyst changes.

**Collaborators:**

- History

**Class:** Validator

**Description:** Responsible for validating and cleansing the content of the user-provided file(s).

**Responsibilities:**

1. Once the client has provided the scripts for the cleansing and validating of the log file data, this class will implement the script against the user files.
2. Determines what errors are found in the user-provided files.
3. Determines what actions the user must take in order to make the log file data valid.

**Collaborators:**

- LogFile

**Class:** Settings

**Description:** Responsible for housing global variables of a given event/session.

**Responsibility:**

1. Keeps track of lead status check.
2. Keeps track of host machine IP address.
3. Keeps track of Red, Blue, and White team directories.
4. Keeps track of root and icon directories.
5. Signals what window to display lead or analyst.

**Collaborators:**

**Class:** ExportTable

**Description:** Responsible for the formatting of the log entry tables that compose the vectors of a given event into an exportable file.

**Responsibilities:**

1. Communicates with the Vector class to obtain the vector names and descriptions.
2. Communicates with the Log Entry class to obtain the log entry identification numbers.
3. Communicates with the Relationship class to obtain the relationship identification numbers.
4. Communicates with the Node class to obtain the Node identification numbers.
5. Communicates with Event to get event details.
6. Compiles retrieved information into a single, exportable file.

**Collaborators:**

- Vector
- LogEntry
- Relationship
- Node
- Event

**Class:** ExportGraph

**Description:** Responsible for the formatting of the final graph with its corresponding vectors into an exportable image file.

1. Renders an exportable image file of the Graph.

**Collaborators:**

- Graph

CRC	Keikaku 企画	Date 2/18/2020 2:39 AM	Page 7
-----	------------	---------------------------	-----------



[END]