**Use Case Scenario Name**: Modify Vector DB

**Description:** Workflow an analyst goes through to modify the global vector DB that includes log file ingestion, log entry association, and local vector DB modification.

**Actors:** Analyst, Splunk, OCR Tool, Transcription Tool, Mongo Db

**Pre-condition:**

1. The PICK Tool server must be running

**Post-condition:**

1. The analyst’s machine is connected to the lead machine, and thus the server.
2. The analyst’s push table is updated to reflect the pushed vector DB.
3. The server saves the pushed vector DB to a system data store.
4. The lead receives the pushed vector DB and is able to approve, reject, or view the changes

**Flow of events:**

Step 1: The analyst starts the program

Step 2: The system displays the starting configuration to the analyst.

Step 3: The analyst enter’s the lead’s IP address and clicks connect.

Step 4: The system connects the analyst’s machine to the lead machine, and thus the server.

Step 5: The system displays the connected configuration to the analyst.

Step 6: The analyst enters a root directory

Step 7: The analyst starts the ingestion process

Step 8: The system cleans all log files in the provided root directory.

Step 9: The system validates all log files, creating enforcement action reports where appropriate.

Step 10: The system sends all validated textual log files to Splunk.

Step 11: Splunk ingests the textual log files as log entries.

Step 12: The system collects all the log entries parsed using Splunk and stores them in the system’s Mongo Db database.

Step 13: The OCR tool ingests image log files as log entries.  
Step 14: The system collects all the log entries from the OCR tool and stores them in the system’s Mongo Db database.

Step 15: Transcription tool ingests video/audio log files as log entries.  
Step 16: The system collects all the log entries from the transcription tool and stores them in the system’s Mongo Db database.

Step 17: The system informs the analyst that all log files have been ingested.

Step 18: The analyst creates a vector.

Step 19: The system stores the vector in a local data store.

Step 20: The analyst requests log entries by providing search criteria to the system

Step 21: The system provides the analyst with log entries that satisfy the search criteria by retrieving them from the system’s Mongo Db database.

Step 22: The analyst associates some of the provided log entries to the created vector.

Step 23: The analyst requests to inspect the created vector.

Step 24: The system displays a graphical and tabular representation of the vector.

Step 25: The analyst modifies the nodes created from the associations.

Step 26: The system updates the graphical and tabular representation of the vector.

Step 27: The analyst defines relationships between nodes.

Step 28: The system updates the graphical and tabular representation of the vector.

Step 29: The analyst modifies the relationships between nodes.

Step 20: The system updates the graphical and tabular representation of the vector.

Step 31: The analyst pushes the local vector DB.

Step 32: End of use case.

**Alt:** Step 3: The analyst enters an invalid IP address.

Step 3.1: The system displays an error message.

Step 3.2: Use case continues at step 2.

**Alt:** Step 6: The analyst enters a non-existent root directory.

Step 6.1: The system displays an error message.

Step 6.2: Use case continues at step 5.

**Alt:** Step 6: The root directory the analyst entered is not formatted correctly.

Step 6.1: The system displays an error message.

Step 6.2: Use case continues at step 5.

**Alt:** Step 7: The analyst chooses to not ingest log files.

Step 7.1: Use case continues at step 11.

**Alt:** Step 18: The analyst chooses to use an existing vector.

Step 18.1: The analyst requests log entries by providing search criteria to the system

Step 18.2: The system provides the analyst with log entries that satisfy the search criteria.

Step 18.3: The analyst associates some of the provided log entries to a chosen vector.

Step 18.4: The analyst requests to inspect the chosen vector.

Step 18.5: Use case continues at step 17.

**Alt:** Step 25: The analyst chooses not to modify the nodes.

Step 25.1: Use case continues at step 20.

**Alt:** Step 25: The analyst chooses not to define relationships between nodes.

Step 25.1: Use case continues at step 24.

**Alt:** Step 29: The analyst chooses not to modify relationships between nodes.

Step 29.1: Use case continues at step 24.

**Alt:** Step 31: The analyst chooses not to push the local vector DB.

Step 31.1: End of use case.