**CRC Assignment**

**Responsibilities & Collaborations**

|  |  |
| --- | --- |
| **Class:** Event | |
| **Description:** Create and save the event settings for an adversarial assessment event. | |
| **Responsibilities**   1. Know the event’s name. 2. Know the event’s description. 3. Know the event’s start date. 4. Know the event’s start time. 5. Know the event's end date. 6. Know the event's end time. | **Collaborations** |

|  |  |
| --- | --- |
| **Class:** Team | |
| **Description:** Setup the Lead environment for the specified event. | |
| **Responsibilities**   1. Know the lead host machine where the master vector DB is stored. 2. Know the number of established connections. | **Collaborations** |

|  |  |
| --- | --- |
| **Class:** Icon | |
| **Description:** Manage icons to be used within the graphing component of the system. | |
| **Responsibilities**   1. Know icon names. 2. Know icon sources. 3. Know icon type. 4. Add new icons. 5. Update existing icon fields. 6. Delete selected icons. | **Collaborations** |

|  |  |
| --- | --- |
| **Class:** Directory | |
| **Description:** Manage the directories that will be read from to retrieve log files | |
| **Responsibilities**   1. Know the directories that will be needed to retrieve log files. 2. Obtain files in known directories. | **Collaborations** |

|  |  |
| --- | --- |
| **Class:** Vector | |
| **Description:** A vector is an important sequence of actions that have been observed/recorded | |
| **Responsibilities**   1. Know vector names. 2. Know vector descriptions. 3. Know which log entries are associated with different vectors. 4. Associate log entries to given vector. 5. Update vector. 6. Delete existing vectors along with all of its associations. | **Collaborations**   * **19 -> Log Entries: 24** * **19 -> Log Entries: 25** * **19 -> Log Entries: 26** * **19 -> Log Entries: 27** * **19 -> Log Entries: 28** * **19 -> Log Entries: 29** * **19 -> Log Entries: 30** |

|  |  |
| --- | --- |
| **Class:** Log Entries | |
| **Description:** Normalized data that was extracted from Splunk | |
| **Responsibilities**   1. Know originating log file creator 2. Know originating log file path 3. Know timestamps 4. Know originating team 5. Know description 6. Know time of occurrence 7. Know the event type. | **Collaborations**   * **23 -> Directory: 15** * **24 -> Directory: 15** * **25 -> Directory: 15** * **26 -> Directory: 15** * **28 -> Directory: 15** * **29 -> Directory: 15** |

|  |  |
| --- | --- |
| **Class:** Cleansing | |
| **Description:** Sends log files to transcriptions if needed and cleanses the files. | |
| **Responsibilities**   1. Send log files to Audio Transcription Tool. 2. Send log files to OCR Transcription Tool. 3. Know cleansing script. 4. Run cleansing script. 5. Know cleansing status. 6. Know cleansing errors. | **Collaborations**   * **30 -> Directory: 16** * **31 -> Directory: 16** * **31 -> OCR Feeder: 101** * **30 -> Parlatype: 103** |

|  |  |
| --- | --- |
| **Class:** Validation | |
| **Description:** Validates log files that have been cleansed. | |
| **Responsibilities**   1. Validate that the cleansed log file content falls within the start date/time. 2. Validate that the cleansed log file content falls within the end date/time. 3. Validate that the cleansed log file content does not contain any empty rows. 4. Approve Validation. 5. Deny Validation. 6. Know Validation errors. 7. Know Validation status. 8. Begin SPLUNK ingestion by sending validated log file. | **Collaborations**   * **36 -> Cleansing: 34** * **37 -> Cleansing: 34** * **38 -> Cleansing: 34** |

|  |  |
| --- | --- |
| **Class:** Enforcement Action Report | |
| **Description:** References the status of log files and their error descriptions. Has the option to cancel the ingestion of selected log files. | |
| **Responsibilities**   1. Display log files that could not be ingested. 2. Display error descriptions for log files. 3. Cancel Ingestion | **Collaborations**   * **44 -> Cleansing: 35** * **44 -> Validation: 41** * **45 -> Cleansing: 35** * **45 -> Validation: 41** |

|  |  |
| --- | --- |
| **Class:** Node | |
| **Description:** Is a graphical representation of a log entry that has been assigned to a vector. | |
| **Responsibilities**   1. Know node id. 2. Know node name. 3. Know node creation time. 4. Know node creation date. 5. Know node long description. 6. Know originating log entry. 7. Know the vector it belongs to. 8. Know Icon Type. | **Collaborations**   * **51 -> Log Entries: 27** * **52 -> Directory: 15** * **54 -> Icon: 9** * **54 -> Icon: 10** * **54 -> Icon: 11** |

|  |  |
| --- | --- |
| **Class:** Graph | |
| **Description:** Displays nodes in either graphical or tabular view. | |
| **Responsibilities**   1. Display node id. 2. Display node name. 3. Display node time of occurrence. 4. Display node long description. 5. Display log entry reference. 6. Display log creator. 7. Display node event type. 8. Hide node id. 9. Hide node name. 10. Hide node time of occurrence. 11. Hide node long description. 12. Hide log entry reference. 13. Hide log creator. 14. Hide node event type. 15. Hide node artifact. 16. Display all node properties. 17. Hide all node properties. 18. Display vector name. 19. Display vector description 20. Display graph in vertical position. 21. Display graph in horizontal position. 22. Zoom in and zoom out into the graph 23. Will have the ability to export a graph in a selected format. 24. Display the graph in the selected time interval units. 25. Know relationships for each vector. 26. Know nodes for each vector. 27. Add blank node to the graph. 28. Remove node from the graph. 29. Edit node. 30. Commit. | **Collaborations**   * **55 -> Node: 47** * **56 -> Node: 48** * **57 -> Log Entry: 28** * **58 -> Log Entry: 27** * **59 -> Log Entry: 24** * **60 -> Log Entry: 23** * **61 -> Log Entry: 29** * **72 -> Vector: 17** * **73 -> Vector 18** * **79 -> Relationship: 85** * **79 -> Relationship: 86** * **79 -> Relationship: 87** * **79 -> Relationship: 88** * **80 -> Vector: 19** * **84 -> Commit: 91** |

|  |  |
| --- | --- |
| **Class:** Relationship | |
| **Description:** Establishes the relationship between two nodes. | |
| **Responsibilities**   1. Know relationship identity number. 2. Know parent node 3. Know child node 4. Know label for the relationship | **Collaborations**   * **86 -> Node: 47** * **87 -> Node:47** |

|  |  |
| --- | --- |
| **Class:** Commit | |
| **Description:** Will keep a record of all the changes made | |
| **Responsibilities**   1. Know changes to be committed 2. Confirm all changes will be updated within the previous version. 3. Commit. | **Collaborations**   * **90 -> Team: 7** * **91 -> Team: 7** |

|  |  |
| --- | --- |
| **Class:** SPLUNK | |
| **Description:** Will store log entries. | |
| **Responsibilities**   1. Transform log files to log entries. 2. Store log entries. 3. Receive log files. 4. Send log entries. | **Collaborations**   * **92 -> Validation: 43** |

|  |  |
| --- | --- |
| **Class:** GraphViz | |
| **Description:** Will display graph | |
| **Responsibilities**   1. Display nodes. 2. Display connectors 3. Display icons. 4. Export as selected format. | **Collaborations**   * **96 -> Node: 47** * **97 -> Relationship: 85** * **98 -> Icon: 10** |

|  |  |
| --- | --- |
| **Class:** OCRFeeder | |
| **Description:** Will transcribe images and PDF files to text files. | |
| **Responsibilities**   1. Transcribe log files received. 2. Send transcribed files back. | **Collaborations**   * **100 -> Cleansing: 31** |

|  |  |
| --- | --- |
| **Class:** Parlatype | |
| **Description:** Audio Transcription Tool. | |
| **Responsibilities**   1. Transcribe mp3 files to text files. 2. Send transcribed files back | **Collaborations**   * **102 -> Cleansing: 30** |