Team404

Prevent, Mitigate, and Recover (PMR) Insight Collective Knowledge System (PICK) Tool Software Design Document

Version 2.0

31 March 2020

Document Control

Approval

The Guidance Team and the customer shall approve this document.

Document Change Control

Initial Release:	0.1
Current Release:	2.0
Indicator of Last Page in Document:	$
Date of Last Review:	30 March 2020
Date of Next Review:	03 April 2020
Target Date for Next Update:	03 April 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. Salamah Salamah
- Dr. Steven Roach
- Elsa Tai Ramirez
- Peter Hanson
- Jake Lasley

Customer:
- Dr. Oscar Perez
- Vincent Fonseca
- Herandy Denisse Vazquez
- Baltazar Santaella
- Florencia Larsen
- Erick De Nava

Software Team Members:
- Eduardo A Jiménez Todd
- Jacob N Torres
- Jorge I Felix
- Alejandro Zamora
- Matthew S Montoya

Change Summary

The following table details changes made between versions of this document

Version	Date	Modifier	Description
0.1	26 February 2020	Matt Montoya	Creation of Document
0.2	06 March 2020	Alex Zamora	Uploaded Database Schema based on ER diagrams by Jorge & Matt
0.3	07 March 2020	Jorge Felix	Finalized Collaboration Diagram with the help of Eddy & Jacob
0.4	08 March 2020	Eddy Todd	Finished Introduction with the help of Jacob & Matt

0.5	09 March 2020	Jacob Torres	Fixed Collaboration Diagram
1.0	09 March 2020	Matt Montoya	Fixed grammar errors, updated DB schema
1.1	22 March 2020	Matt Montoya	Updated *Maltego Class* Contract
1.2	22 March 2020	Matt Montoya	Updated *Log File Class* Contract
1.3	22 March 2020	Matt Montoya	Updated *Significant Log Entry Class* Contract
1.4	23 March 2020	Jacob Torres	Updated *Graph Class* Contract
1.5	24 March 2020	Alex Zamora	Updated *Vector Class* Contract
1.6	24 March 2020	Eddy Todd	Updated *OCR Class* Contract
1.7	24 March 2020	Eddy Todd	Updated *Event Configuration Class* Contract
1.8	24 March 2020	Eddy Todd	Fixed *Significant Log Entry Class* Contract
1.9	24 March 2020	Alex Zamora	Updated *Enforcement Action Report Class* Contract
1.10	24 March 2020	Alex Zamora	Updated Icon Class Contract
1.11	24 March 2020	Jorge Felix	Updated *Directory Class* Contract
1.12	24 March 2020	Jorge Felix	Updated *Search Class* Contract
1.13	25 March 2020	Matt Montoya	Added new introductions to Sections 2, 3, 4; Fixed grammar errors
2.0	30 March 2020	Alex Zamora, Matt Montoya, Jorge Felix, Jacob Torres	Fixed issues based on TA feedback

Table of Contents

1. Introduction

Section 1 shall introduce Team404's Software Design Document (SDD) document for the Spring 2020 Software II Project, *PICK Tool*, including the purpose and intended audience, overview, document references, as well as definitions, acronyms, and abbreviations.

1.1. Purpose and Intended Audience

The purpose of creating the software design document is to describe the Prevent, Mitigate Recover (PMR) Insight Collective Knowledge (PICK) Tool, providing the software development team guidance for the architecture of the project. This SDD details the specifications of the characteristics of the design components. The description of the design is necessary for Team404 to work under a single, guided vision of what PICK Tool shall be. Furthermore, the SDD shall act as a point of reference, outlining all parts of the software and the interactions between them.

1.2. Scope of Product

PICK Tool shall facilitate the job of analysts during an Adversarial Assessment of a simulated cyber-attack, reducing the time it currently takes analysts to perform an assessment to about two weeks. In doing so, PICK Tool shall assist analysts in telling the true story pertaining to these simulated attacks. To help satisfy these needs, analysts will utilize PICK Tool to search through and filter through logs, or recorded notes from the systems of attackers and defenders, as they may pertain to a simulated attack. Though PICK Tool, analysts can use PICK Tool to construct a vector or visual graph of events that satisfy an objective. More information regarding logs can be found in Section 2 of this document.

1.3. References

[1] Wirfs-Brock, R., Wilkerson, B. and Wiener, L. (1990). Designing object-oriented software. Englewood Cliffs, N.J.: Prentice-Hall.

1.4. Definitions, Acronyms, and Abbreviations

1.4.1. Definitions

TERM	DEFINITION
Active Scene	The scene that is currently displayed.
Actor	A user or external system that interacts with the system in the use case diagram.
Adversarial Assessment	Analysis of a simulated cyberattack by the White Team.
Analyst	The Analyst is the primary user of PICK Tool. Multiple Analysts can access PICK Tool simultaneously. The analyst primarily uses PICK Tool to ingest log files, correlate logs, and create graphs within the system.
Associated Log	Logs with a cause and consequence log linked together.
Audio Transcription Tool	A software program that takes audio logs and transcribes them into text files.
Blue Team	Defenders during the simulated cyber-attack; the team that will defend their system from attack.
Client(s)/Customer(s)	The U.S. Army Combat Capabilities Development Command Data & Analysis Center: Lethality, Survivability & Human System Integration (LSH) Directorate; individuals from this directorate include Dr. Oscar Perez, Mr. Vincent Fonseca, Mr. Baltazar Santaella, Ms. Herandy Vazquez, Ms. Florencia Larsen, & Mr. Erick De Nava.
Client-Server Model	PICK Tool running on a server in a closed system.
Commit	A button that confirms the actions of the analyst to save the project.
Correlated Logs	Two log entries connected through cause and effect by the analyst.
Database	A structured set of data held in a computer, especially one that is accessible in various ways.
ETL Tool/Log Management Tool	Software that combines the Extract, Transform, Load database functions into one tool to take data from one or many sources into a destination system.
Event Log	A detailed record of system events stored in the System Event Viewer by the computer's operating system.
Filter	A way for the analyst to search for specific queries through specific conditions set by the analyst.
Filter Space	A certain condition set by the analyst.
Formatted Log Entry	A sanitized log entry that has been cleaned up based on a set configuration by the analyst.
Graph	A visual representation of the scenario between the BlueTeam and the Red Team shown through nodes and vectors. [1]
Graphing Tool	A software program that will construct the graph based on the vectors and correlations made by the analyst.
Graphical User Interface	The way that the system will display all components of the system to the user, the analyst.
Ingest/Ingestion	When files, specifically formatted log files, are put into our system to be correlated, edited and disregarded if need be.
Kali Linux	Operating System the clients will use.
Local Network	A data communications network within the system.
Log	An official record of events.

Log Entry	A log entry is the output of the log file. Log entries contain information from the log file and are recorded details of an ingested log.
Log File	A file is an input into the system that records either events that occur in an operating system or other software runs or messages between different users of communication software.
Natural Language Processor	An external software program that takes audio files and translates them into text files to be later validated.
Node	A node is a visual representation of a significant event that was marked for the current vector.
Normalize	To make rid of duplicate entries.
Objective	The goal of the attackers when they attack a system. It is the goal of the defenders to prevent attackers from achieving their goal(s).
Observer	A member of the white team that takes notes and assessments during the staged cyber-attack between the Blue Team and the Red Team.
Optical Character Reader	An external software program that takes image files and translates them into text files to later be validated.
PICK Tool	The software product which Team404 is tasked with constructing for the clients.
Red Team	The title given to the attackers in the simulated cyber-attack; The team that will attack the system the blue team will defend.
Sanitized Logs/Sanitized Log Entries	A log entry that has gone through validation.
Significant Event	Logs that are important to the overall cyber-attack scenario between the Blue Team and the Red Team.
Team404	CS4311 Team 6, the software development team; this includes Mr. Alejandro Zamora, Mr. Eduardo Jiménez Todd, Mr. Jacob Torres, Mr. Jorge Felix, and Mr. Matt Montoya.
Timeline	A set of events, logs, that help convey the overall story of the events that took place between the Blue Team and the Red Team.
Timestamp	The time that a log was recorded in the system during the simulated cyber-attack.
User/Users	A person that interacts with the system; in PICK, the user will be an *Analyst*
Vector	The series of activities/steps an adversary executes or attempts to execute that is necessary to achieve an objective.
White Team	The title given to the observers during the simulated cyber-attack. They observe what happens between the Blue Team and the Red Team on the system.
Wildcard	Unsearchable characters, and any characters that can be searched through.
Zulu Time	The military and navigation parlance for the UTC time standard.

1.4.2. Acronyms

TERM	**DEFINITION**
AA	Adversarial Assessment
AKA	As Known As
CSV	Comma Separated Value.
DOD	Department of Defense.
ETL	Extract Transform Load.
GUI	Graphical User Interface

IEEE	Institute of Electrical and Electronics Engineers
JPG/JPEG	Joint Photographic Experts Group.
NLP	Natural Language Processor.
OCR	Optical Character Reader.
OS	Operating System
PDF	Portable Document Format.
PICK	PMR Insight Collective Knowledge.
PMR	Prevent, Mitigate Recover.
SDD	Software Design Document

1.4.3. Abbreviations

Admin	Administrator

1.5. Overview

1.5.1. Introduction

Section 1 shall introduce Team404's Software Design Document (SDD) document for the Spring 2020 Software II Project, *PICK Tool*, including the purpose and intended audience, overview, document references, as well as definitions, acronyms, and abbreviations.

1.5.2. Decomposition Description

Section 2 (Decomposition Description) introduces the System Collaboration Diagram (SCD), the Subsystem and Component Descriptions (CRC Cards), as well as the dependencies of the system, defining the responsibilities entities have for specific functions, and tracing system requirements to design entities.

1.5.3. Detailed Description of Components

Section 3 (Detailed Description of Components) provides complete, detailed descriptions of all components listed within the System Collaboration Diagram (Section 2.1 of the SDD). These components include external as internal components, as they relate to PICK Tool.

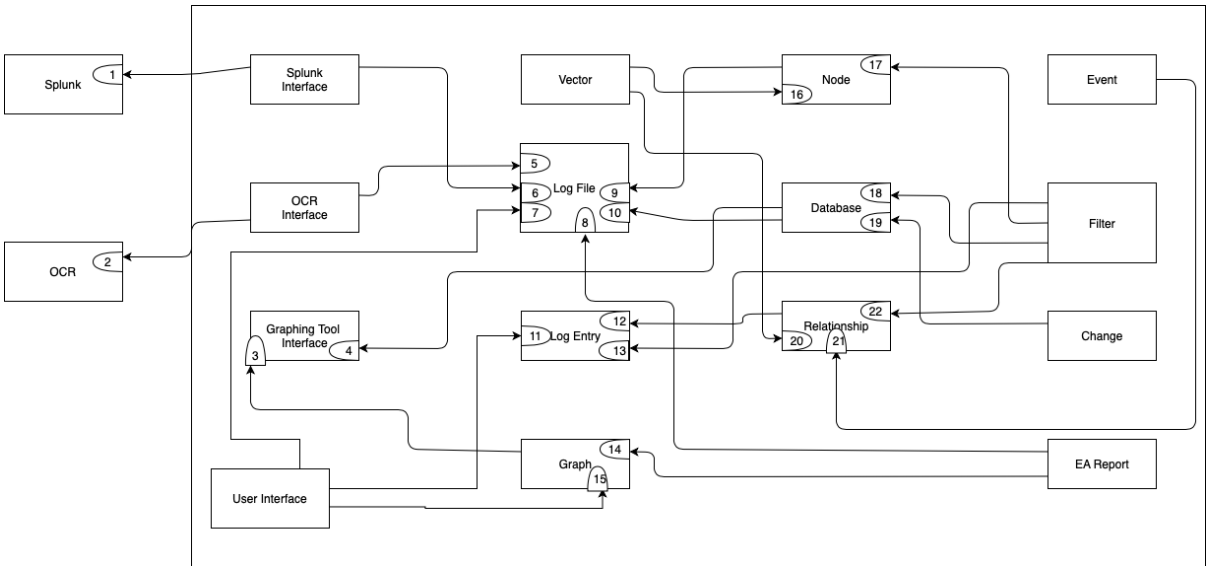
1.5.4. Database

Section 4 introduces the database (DB) that will be used for PICK Tool. This includes the Entity-Relationship (ER) Diagram and database schema that represent the graphical representation of entities and logical configuration of the relational database, respectively.

2. Decomposition Description

Section 2 introduces the System Collaboration Diagram (SCD), the Subsystem and Component Descriptions (CRC Cards), as well as the dependencies of the system, defining the responsibilities entities have for specific functions, and tracing system requirements to design entities.

2.1. System Collaboration Diagram



2.2. Subsystem and Component Descriptions

Splunk Interface - Communicates with Splunk

OCR Interface - Communicates with OCR

Graphing Tool Interface - third party software tool for graphing that communicates with graph

Vector - A description of a significant event communicates with nodes and relationships

Log File - Computer data object that stores logs

Log Entry - Logs that are important to the system to the overall scenario

Graph - Visual representation of a vector communicates with graphing tool interface

Node - Visual representation of a significant event communicates with log file

Database - Stores cleansed logs of vectors ingested communicates with log file

Relationship - stores details on relations between nodes

Filter - Allows the user to search for specific criteria

Change - Path and structure of the log files

Event Action Report(EA) - Handles errors

Splunk - Handles log ingestions and cleansing

OCR - recognize characters in images

User Interface - what the user interacts with.

2.3. Dependencies

Dependencies are external elements to the program. These dependencies are required by the program in order to work properly.

- **Kali Linux** - Kali Linux is a Debian GNU / Linux based distribution designed primarily for general computer security and auditing.
- **Python v3.6.7 or later** - Python is an interpreted programming language whose philosophy emphasizes the readability of its code. It is an interpreted, dynamic and cross platform language.
- **Splunk** - Splunk is a software to search, monitor and analyze big data generated through a web interface.
- **OCR Tool** - Optical Character Recognition is a process aimed at digitizing texts, which are automatically identified from an image symbols or characters that belong to a certain alphabet.
- **MongoDB** - MongoDB is a NoSQL database system. Instead of storing data in tables, just as you do in relational databases, MongoDB saves BSON data structures making data integration easier and faster.
- **PyQt5** - PyQt is a binding of the Qt graphics library for the Python programming language.
- **TKinter** - Tkinter is a binding of a graphics library for the Python programming language. It is considered a standard for the graphical user interface for Python.

3. Detailed Description of Component <name>

Section 3 provides complete, detailed descriptions of all components listed within the System Collaboration Diagram (Section 2.1 of the SDD). These components include external components such as the Database and Splunk, as well as internal components including, but not limited to, the OCR Interface and Graphing Tool Interface.

3.1. Splunk Interface

Class: Splunk Interface	
Superclass: None	
Subclasses: None	
Description: A third party software tool for transforming log files into normalized log entries.	
(1) Ingestion Contract	
Responsibilities: 1. Can ingest log files. 2. Can transform a log file into normalized log entries. 3. Can store log entries in the normalized data files. 4. Can process incoming data into individual activities according to the nature of the data. 5. Can export log entries	**Collaborations:** Log Files (6) Splunk (1)

3.2. OCR Interface

Class: OCR
Superclass: None
Subclasses: None
Description: Allows the user to make any type of document into a word-searchable document.
(2) OCR Knows Elements

Responsibilities: 1. Can import files 2. Can read documents 3. Can recognize character 4. Can translate images to text 5. Can translate text to a .txt document 6. Can export files	**Collaborations:** Log Files (5) OCR (2)

3.3. Graphing Tool Interface

Class: Graphing Tool Interface	
Superclass: None	
Subclasses: None	
Description: A third party software tool for graphing.	
(3) Create Graph	
Responsibilities: 1. Knows export format. 2. Can create graphs with nodes and relationships between nodes. 3. Can export graph.	**Collaborations:**

3.4. Vector

Class: Vector	
Superclass: None	
Subclasses: None	
Description: A description of a significant event.	
(4) Vector Elements	
Responsibilities: 1. Knows Vector Name 2. Knows Vector Description	**Collaborations:** Nodes (16) Relationship (20)

3. Knows Vector Entries	

3.5. Log File

Class: Log File	
Superclass: None	
Subclasses: None	
Description: A computer data object that stores logs.	
(5) Splunk Integration	
Responsibilities: 1. Can store the following components:(splunk) a. Log File Name b. Cleansing Status c. Validation Status d. Ingestion Status e. Acknowledgment, as well Status 2. Can export log files	**Collaborations:**
(6) OCR Integration	
Responsibilities: 1. Can contain extracted text if the log file is of type “image” or “pdf”.(OCR)	**Collaborations:**
(7) Node Integration	
Responsibilities: 1. Can contain a timestamp per line, bounded by the start date, end date, start time, and end time specified in the event configuration. (Node)	**Collaborations:**
(8) Database Integration	
Responsibilities: 1. Can contain transcription with timestamps in one-minute intervals if the log file is of type “audio” or “video”. (Database)	**Collaborations:**

(9) EA Report Integration	
Responsibilities: 1. When a timestamp property of a previously saved event is changed, the impact of the change shall be restricted to the "not-validated" log files. (EA Report)	**Collaborations:**

3.6. Log Entry

Class: Significant Log Entry	
Superclass: Log Entry	
Subclasses: None	
Description: Logs that are important to the overall cyber-attack scenario between the Blue Team and the Red Team.	
(10) Relational	
Responsibilities: 1. Knows relevant information for a vector.	**Collaborations:**
(11) Filter Log Entry	
Responsibilities: 2. Knows the following: a. Log Entry Number b. Log Entry Timestamp c. Log Entry Content d. Host e. Source f. Source Type	**Collaborations:**

3.7. Graph

Class: Graph
Superclass: None.
Subclasses: None.

Description: A visual representation of a vector.	
(10) Graph Action Report	
Responsibilities: 1. A graph shall comprise at least one node. 2. Can allow the analyst to add nodes, edit nodes, delete nodes, add relationships, edit relationships, and delete relationships. 3. Can create a PNG of the graph. 4. Knows how to export format: 5. Knows orientation 6. Knows Vector Name 7. internal units 8. Knows interval 9. Knows position of nodes 10. Knows position of relationships	**Collaborations:** graphing tool interface(3)

3.8.Node

Class: Node	
Superclass: None	
Subclasses: None	
Description: Visual representation of a significant event that was marked for the current vector.	
(11) Node Elements	
Responsibilities: 1. Can represent a log entry as part of the graph 2. Knows the following: a. Node ID b. Node Name c. Node Timestamp d. Node Description e. Log Entry Reference f. Log Creator g. Event Type h. Icon Type i. Source j. Node Visibility	**Collaborations:** logfile(9)
(12) Vector Element	
Responsibilities:	**Collaborations:**

1. Can provide information to the graph. 2. Can link informations 3. Knows information of Icon	
(13) Filter Elements	
Responsibilities: 1. Knows the following: k. Node ID l. Node Name m. Node Timestamp n. Node Description o. Log Entry Reference p. Log Creator q. Event Type r. Icon Type s. Source t. Node Visibility	**Collaborations:**

3.9. Database

Class: Database	
Superclass: None	
Subclasses: None	
Description: A database that shows cleansed logs of vectors ingested.	
(14) Graph Tool	
Responsibilities: 1. Can notify the lead when a record is pushed. 2. Can allow the lead to approve pushes.	**Collaborations:**
(15) Database Elements	
Responsibilities: 1. Knows the cleansed log files in permanent storage. 2. Can store significant log entries in permanent storage.	**Collaborations:** logfile(10)

(16) Filter elements	
Responsibilities: 1. Can get vectors from analysts.	**Collaborations:**
(17) Change Tracker	
Responsibilities: 1. Knows the Changes made to log files in permanent storage. 2. Can store changes of log entries in permanent storage.	**Collaborations:** logfile(10)

3.10. Relationship

Class: Relationship	
Superclass: None	
Subclasses: None	
Description: Represent the relationship between nodes.	
(18) Vector Relationships	
Responsibilities: 1. Knows the relationship ID 2. Knows the parent ID 3. Knows the child ID 4. Knows the label	**Collaborations:** log entry(12)
(19) Event Parameters	
Responsibilities: 1. Can correlate nodes to one another.	**Collaborations:**

3.11. Filter

Class: Filter	
Superclass: None	
Subclasses: None	
Description: Gives the user to search for specific criteria	
(22) Seeks	
Responsibilities: 1. When a search operation is complete, the system can return a result that matches the searched keyword with the searched keyword highlighted in the search result. 2. Can perform logical searching. 3. Can perform wildcard searching.	**Collaborations:** Node(17) database(18) log entry(13) relationship(22)

3.12. Change

Class: Change	
Superclass: None	
Subclasses: None	
Description: Path and structure of the log files.	
(23) Keeps Track of	
Responsibilities: 1. Can perform structure check when data ingestion starts. 2. Can generate root directory structure error if it fails to contain three folders or if the folder names specified in the event configuration don't match. 3. Can store the log files in their corresponding folder. 4. Knows the following: a. Red Team Folder b. Blue Team Folder c. White Team Folder	**Collaborations:** database(19)

3.13. Event Action (EA) Report

Class: Enforcement Action Report	
Superclass: None	
Subclasses: None	
Description: Handles errors when anomalous events happen.	
(24) EA Elements	
Responsibilities: 1. Can track error codes 1. Can display error code 2. Can track description of error codes 3. Can track the state of the system 4. Knows the Line Number 5. Knows the Error Message	**Collaborations:** log file(8) graph(14)

3.14 Splunk

Class: splunk	
Superclass: None	
Subclasses: None	
Description:Handels log ingestion and cleansing.	
(25) log handling	
Responsibilities: 1. handles log cleansing 2. handles log ingestion	**Collaborations:**

3.15 OCR

Class: OCR
Superclass: None
Subclasses: None
Description: recognizes character in images .

(26) image process	
Responsibilities: 1. recognize text in images	**Collaborations:**

3.16 user interface

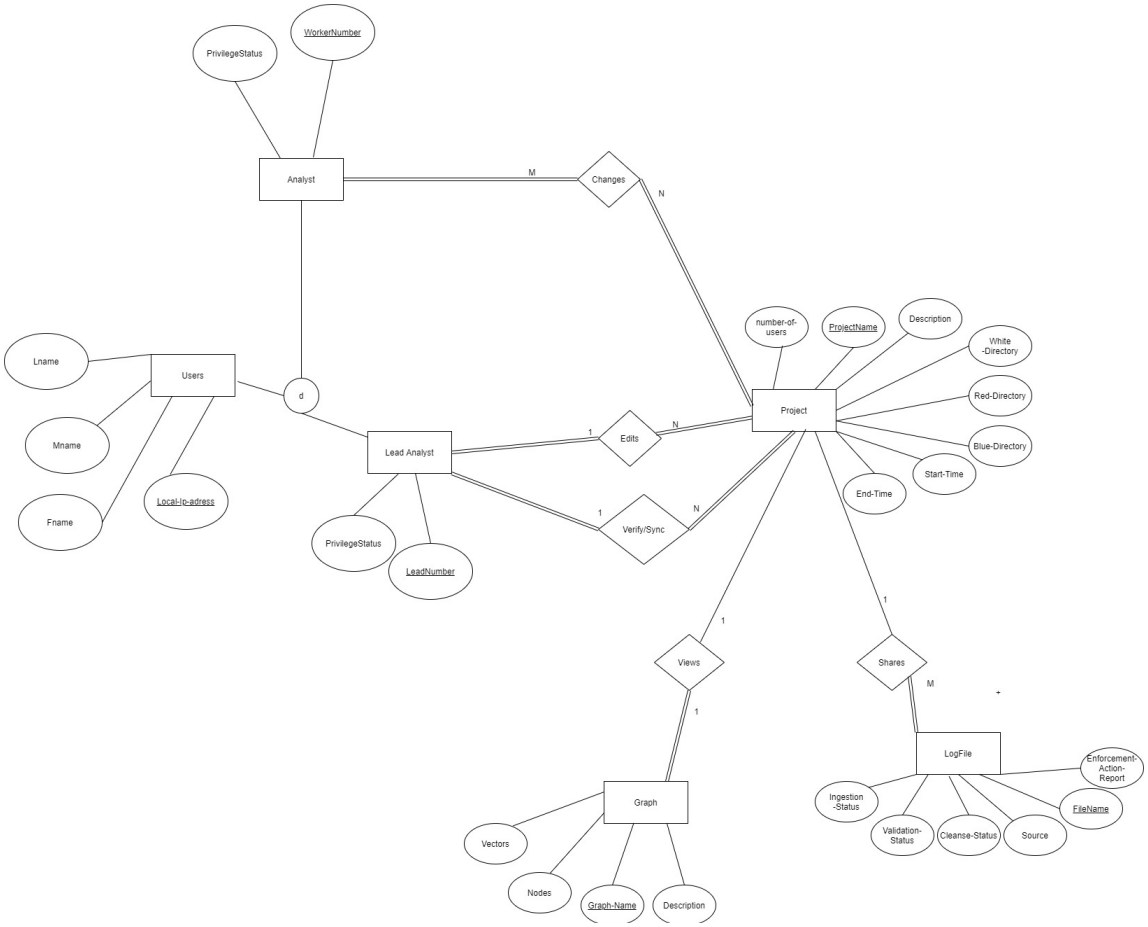
Class: user interface	
Superclass: None	
Subclasses: None	
Description: what the user sees and interacts with.	
(27) display	
Responsibilities: 1. display current window user is interacting with 2. keep track of parallel tasks	**Collaborations:** log file(7) log entry(11) graph(15)

4. Database

Section 4 introduces the database (DB) that will be used for PICK Tool. This includes the Entity-Relationship (ER) Diagram and database schema that represent the graphical representation of entities and logical configuration of the relational database, respectively.

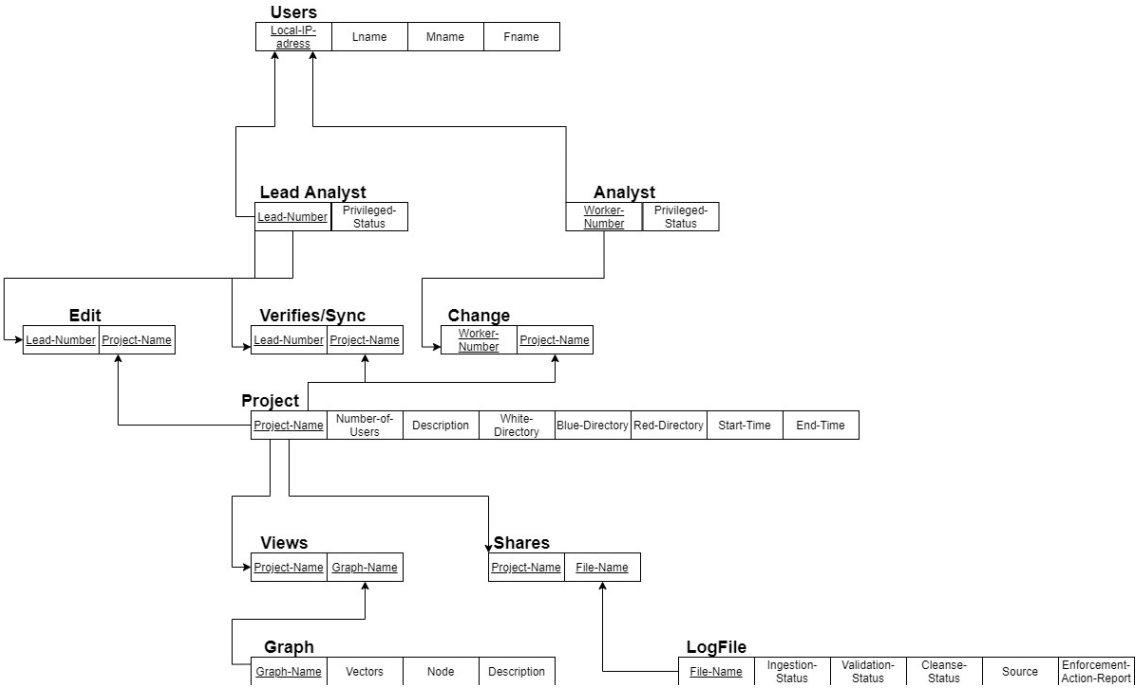
4.1. Entity-Relationship Diagram

The ER diagram, shown below, is a conceptual design of the relationship between entities, or things in the real-world as they pertain to PICK Tool. These entities are denoted by the rectangular shapes in the model. These entities may contain attributes that describe the entities, as well as relationships between entities. Attributes are housed within the oval shapes, where relationships are housed within the diamond shapes; both of which are connected to the entities.



4.2. Database Schema

The Database Schema (or relational model) shown below, is another visual representation of the database, depicting the logical configuration of the database, including the set of rules that govern as they pertain to PICK Tool. This schema serves as a description of the database itself and is derived from the ER Diagram in Section 4.1. The conversion of high-level (ER diagram) to logical design (DB Schema) follows a seven-step algorithm: Mapping regular entities to relations, mapping weak entities to relations, identifying 1:1 binary relationship types, identifying binary 1:N relationship types, identifying M:N relationship types, separating multi-valued attributes, and creating new relations for N-ary relationships.


Users
Local-IP-adress
Lname
Mname
Fname
Lead Analyst
Lead-Number
Privileged-Status
Analyst
Worker-Number
Privileged-Status
Edit
Lead-Number
Project-Name
Verifies/Sync
Lead-Number
Project-Name
Change
Worker-Number
Project-Name
Project
Project-Name
Number-of-Users
Description
White-Directory
Blue-Directory
Red-Directory
Start-Time
End-Time
Views
Project-Name
Graph-Name
Shares
Project-Name
File-Name
Graph
Graph-Name
Vectors
Node
Description
LogFile
File-Name
Ingestion-Status
Validation-Status
Cleanse-Status
Source
Enforcement-Action-Report

$