<<Git Guys>>

<Finding Reporting Information Console>

Software Requirements Specification

Version <0.14>

<May 17, 2020>

Document Control

Approval

The Guidance Team and the customers will approve this document.

Document Change Control

Initial Release	0.1
Current Release	0.14
Indicator of Last Page in Document	</3
Date of Last Review	5/17/2020
Date of Next Review	5/18/2020
Target Date for Next Update	5/18/2020

Distribution List

This following list of people will receive a copy of this document every time a new version of this document becomes available:

Guidance Team Members: Elsa Tai Ramirez, Ben Robertson

Customer: Cyber Experimentation & Analysis Division (CEAD)

Software Team Members: Alex Vasquez, Isaias Leos, Andrew Clanan, Luis Soto, Jacob Padilla

Change Summary

The following table details changes made between versions of this document

Version	Date	Modifier	Description
0.1	May 7, 2020	Alex Vasquez	Moved over all existing documents into place. Next work on section 1
0.2	May 7, 2020	Alex Vasquez	Fixed document

			hygiene, made sure the sections are labeled correctly
0.3	May 7, 2020	Alex Vasquez	Added in graphs to appendix
0.4	May 7, 2020	Alex Vasquez	Section 1.1, 1.3, 1.4, 1.5
0.5	May 7, 2020	Team Assignment	Address all TODO in comments Individual review to make sure we are not missing anything **VERY LAST THING:** Fix page numbering and table of contents One last review and submit
0.6	May 12, 2020	Jacob Padilla	Added to the introductions
0.7	May 12, 2020	Jacob Padilla	Fixed the document structure
0.8	May 14, 2020	Jacob Padilla	Added to the Use case scenarios
0.9	May 17, 2020	Isaias Leos	Fixed Data flow diagram by removing event and system, also removed unnecessary connection as per feedback
0.10	May 17, 2020	Isaias Leos	Fixed state transition table for data sync, added the images for each diagram
0.11	May 17, 2020	Isaias Leos	Added section 3.2

0.12	May 17, 2020	Isaias Leos	Added missing contents for 3.2.2 and fixed a few to require shall and wording to make better sense.
0.13	May 17, 2020	Andrew Clanan	Fixed/updated use case diagram, and fixed non behavioral requirements Added non behavioral requirement overview. Added references.
0.14	May 17, 2020	Luis Soto	Fixed class diagram relationships. Added intros to diagrams in the appendix.

Table of Contents

1. Introduction

1.1. Purpose and Intended Audience

The purpose of this SRS is to give a complete and concise description of the external interface of the Finding and Reporting Information Console system and its interaction with its environment. This document aims to provide an agreement point for both the Cyber Experimentation and Analysis Division team and the development team concerning the requirements of the external interface of the Finding and Reporting Information Console. This document is only intended to be viewed by the Cyber Experimentation and Analysis Division team and the development team as there is information that is confidential to all parties not mentioned above.

1.2. Scope of Product

The system to be produced is the Finding and Reporting Information Console, commonly known as F.R.I.C. . This Is a tool used by C.E.A.D analysts to keep track of vulnerability attacks while testing a system(s) at an event. This includes uploading artifacts to the console, the ability to create/edit/delete depending on the authorization, notifications on tasks, and backlog tracking on analyst work. The ultimate goal of F.R.I.C. is a tool used to test systems during an event, and to make sure the analyst during the event can properly document the findings related to a task/subtask.

1.3. Definitions, Acronyms, and Abbreviations

This section contains definitions, acronyms, and abbreviations used to aid the reader in complete understanding of the document.

1.3.1. Definitions

This section contains a list of terminology used throughout the document. In Table 1, the following terminology list is defined for the readers of the document.

Table 1:

TERM	DEFINITION

Event	Describes an actual assessment that is typically 5 days long. This is also known as a penetration test. Projects may take 6 months to complete but the actual event is when the team will be hands on with the systems “hacking” away.
Lead Analyst	The lead analyst oversees the Event. The lead analyst has access to everyone's work and a main responsibility is assigning tasks to analysts. The Lead Analyst manages the progress of the entire event
Analyst	An analyst performs tasks, either assigned or picked up, and is responsible for reporting any vulnerability that he/she has found.
System	A system refers to what analysts will be testing for vulnerabilities. A specific portion of the scope given by the client (i.e Wells Fargo), that is being tested. (i.e. Wells Fargo ATM). Analysts will be trying to “crack” vulnerabilities
Task	A job that is assigned by a Lead Analyst to an analyst. In F.R.I.C. It is important to note that only a Lead analyst can assign these jobs to analysts.
Sub Task	Similar to a task in the sense that they are simply jobs to be done, though an analyst has the ability to create a subtask for themselves. A subtask is the child of a task and can only exist under an existing task.
Finding	A finding is a vulnerability that was found in a specific system. Findings can come from doing a task or can be standalone, better known as an orphan finding. A finding can also be informational rather than an actual vulnerability.

1.3.2. Acronyms

This section contains a list of acronyms used throughout the document. In Table 2, the following list of acronyms is defined for the readers of the document along with their definitions.

Table 2:

TERM	DEFINITION
F.R.I.C.	Finding and Reporting Information Console System
C.E.A.D.	Cyber Experimentation and Analysis Division
SRS	Software Requirements Specification

1.3.3. Abbreviations

This section contains a list of abbreviations used throughout the document. In Table 3, the following list of abbreviations is defined for the readers of the document along with their definitions.

Table 3:

TERM	DEFINITION
Sync	Synchronize data. This is used whenever a user of the system is trying to push or pull data in the system.

1.4. Overview

Section 2: General Description:

Section 2.1 Product Perspective

The Product Perspective section describes the product itself and the functionality that it provides. In this section we describe how F.R.I.C aims to encapsulate every step of CEAD's process in a simple and efficient manner.

Section 2.2 Product Features
Product Features is a section which describes the essential features of F.R.I.C. In this section you will find use cases for F.R.I.C along with the use case descriptions. The last part of this section shows different scenarios in which these use cases would be used.

Section 2.3 User Characteristics
The User Characteristics section describes how the users of F.R.I.C, which are the Lead Analyst and Analyst, interact with the system. In this section you will find the difference between the two uses and their role in the system.

Section 2.4 General Constraints
The General Constraints section describes factors that constrain the options of the development team. This section may include security, hardware, safety, and organizational considerations.

Section 2.5 Assumptions and Dependencies
The Assumptions and Dependencies section details the assumptions and dependencies that the F.R.I.C system is built upon. This is the basis for which the team has formulated their requirements for the F.R.I.C system

Section 3: Specific Requirements

Section 3.1 External Interface Requirements
This section contains the specification of requirements for interfaces among different components and their external capabilities, including all its users, both human and other systems. In this section you will find images of the approved prototype and details concerning all information on the given page.

Section 3.2 Behavioral Requirements

The Behavioral Requirements section defines what the system actually does. Inside of this section you will find the following 4 sub sections:

Same Class of User- Describes the accesses and permissions of the different users in the F.R.I.C system

Related Real-World Objects- Describes the objects in the F.R.I.C system and the attributes associated with them. Here you will find the attribute along with a description, the data type, and constraints associated with this attribute.

Stimulus- Describes how the specific part of the system is supposed to react to other interactions in the system. This section is broken down into two parts which are the User

Interface Stimulus and the System Stimulus. The User interface Stimulus describes reactions to interactions with the user interface while the System Stimulus describes the behavior the system exhibits when provided with a system based interaction.

Section 3.3 Non-Behavioral Requirements
The non-behavioral requirements section describes the attribute that describes how F.R.I.C. should perform. The requirements in this section describe the quality attributes that define the quality attribute for the entire scope of the system.

1.5. References

[1] “Interview Report,” GitGuys, El Paso, TX, 2020.

[2] “Software Capstone Project Findings and Reporting Information Console (F.R.I.C.)” unpublished.

[3] “Answered Memo”, GitGuys, Cloud9, Cactus Terror, Team 1,El Paso, TX, 2020.

[4] “Answered Memo 2 ”, Team 2,El Paso, TX, 2020.

[5] “Answered Memo 4 ”,Cactus Terror,El Paso, TX, 2020.

[6] “Answered Memo 5 ”, Hot Java,El Paso, TX, 2020.

[7] “Answered Memo 6 ”, Team 8,Team 13,Team 4 ,Team 10, El Paso, TX, 2020.

[8] “Answered Memo 7 ”, Cactus Terror ,El Paso, TX, 2020.

[9] “Prototype”, GitGuys, El Paso, TX, 2020.

[10] “Risk Matrix Template”, C.E.A.D, El Paso,TX, 2020.

[11] “Traceability Report”, Git Guys, El Paso, TX, 2020.

[12] “Feasibility Report”, Git Guys, El Paso TX, 2020.

2. General Description

2.1. Product Perspective

CEAD is looking for a system to efficiently facilitate information on security vulnerabilities in various softwares and hardwares that they penetrate testing for clients. F.R.I.C is a system in which a team of analysts can organize an entire cyber engagement(s), which is broken down into Events, Tasks, Subtasks, and Findings. The role of the analyst is penetration testing and CEAD would like to have an organized, structured way of managing the entire engagement and viewing the progress throughout this process. Each Lead Analyst will assign unique tasks for the analyst to complete.An event is the entire engagement, and in each event there are multiple systems. The job of the analyst is to test the multiple systems for any vulnerabilities. An example of a task would be to try to find a backdoor entrance in a given system. The sole purpose of these tasks is to try and find a vulnerability, and this is where the report comes to play. If a vulnerability is present, it is labeled as a finding. There are then three reports that come from a cyber engagement. The three reports are the Technical Report, The Emergence Result Briefing, and the Risk Matrix. These three reports give the details of findings in the software or hardware tested and are essential to convey the seriousness of the threat in the systems tested to the client. F.R.I.C aims to encapsulate every step of CEAD's process in a simple and efficient manner by delivering a central repository for the entire engagement.

2.2. Product Features

This section is used to illustrate: level 2 use case diagram, descriptions of actors, and descriptions of use cases. Below we break down all the components that are essential in each section based on the clients needs.

2.2.1. Level 2 Use Case Diagram

In figure 1 it presents the entities participating in the F.R.I.C system (lead analyst, analyst and the database) and the different activities (use cases) F.R.I.C must carry out (F.R.I.C system) represented as a use case diagram.

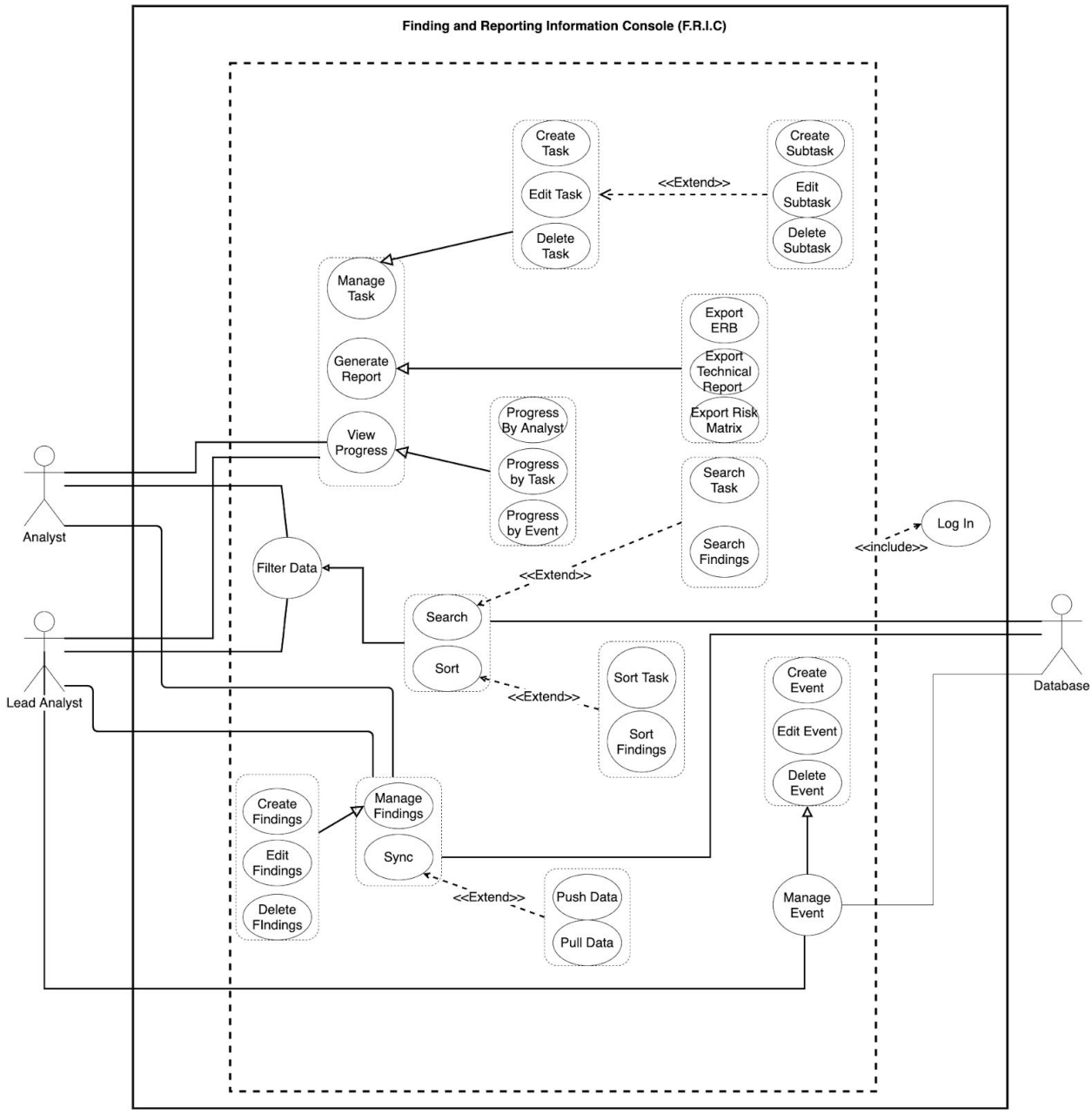


Figure 1: Level 2 Use case diagram.

2.2.2. Description of Actors

The following list presents the actors found in the usage diagram, along with a brief description.

A. Analyst:

Task will be assigned to at least one analyst. Once a vulnerability is confirmed, an analyst will document (Adding a title, host IP, note, attachments, and a mitigation) this finding so it can be added to the technical report when the event is done.

B. Lead Analyst:
   Will work with PM and Subject Matter Experts (SME) to identify cyber vulnerabilities. A lead will assign portions/requirements of a system to each analyst so they can focus on finding vulnerabilities only in their assigned system space. Create/edit/delete systems, tasks, sub-tasks and assign those tasks and sub-tasks to other analysts. Will report findings of the team to PM for each vulnerability.
C. Database:
   The database is in charge of storing information about vulnerabilities of a system, creating a record of what each analyst is working on, holding credentials for each analyst, allowing Lead analyst and analyst to push and pull tasks, sub tasks, findings and progress.

2.2.3. Description of Use Cases

In the following list they will be presented the description of actions (use cases) within the F.R.I.C system:
Use case 1: Action that will allow a lead analyst to identify whether they would like to create, edit, or delete a task
Use case 2: Action that will allow a lead analyst to choose which report they would like to export
Use case 3: The user (Lead Analyst) would like to see how the progress of the cyber engagement
Use case 4: Create a brand new task.
Use case 5: Edit the properties of a task
Use case 6: Delete an established task from the database
Use case 7: Will create and save a subtask which falls under the umbrella of a given task.
Use case 8: Action that will allow a lead analyst or analyst to change the properties of a subtask
Use case 9: Action that will allow the lead analyst and analyst to delete subtasks from the database
Use case 10: Lead Analyst or analyst can select what information they want presented to them from tasks, subtasks, and findings.
Use case 11: Database will organize all tasks, subtasks, and findings for easy finding.
Use case 12: Allows users to look through tasks, subtasks, and findings.
Use case 13: The system uses the database to sort the tasks according to the lead analyst criteria.
Use case 14: Arrange the findings in a way specified by the analyst and/or lead analyst
Use case 15: Allows the user to look for a certain task.
Use case 16:Allows the user to look for a certain finding.
Use case 17: Will compare current data of the analyst and lead analyst to compile the most recent information.
Use case 18:View the findings available to the analyst(s) with the purpose of viewing, editing, or deleting a finding.

Use case 19:Create a record of information found during an analysis.
Use case 20:Action that allows the user to change the properties in a certain finding.
Use case 21: Allow the lead analyst and analyst to delete a finding from the database.
Use case 22: An analyst and lead analyst will use their credentials to login.
Use case 23: System allows any analyst to receive data that has been pushed by another analyst.
Use case 24: System allows any analyst to upload their current data to the main system.
Use case 25: Create a brand new Event
Use case 26: Action that will allow the Lead Analyst to generate a ERB report
Use Case 27: Action that will allow to generate a Technical Reportreport
Use Case 28: Action that will allow to generate a Risk Matrix report
Use Case 29: Lead analyst(s) request to add or change event information
Use Case 30: Lead Analyst(s) will be allowed to delete the entire event
Use Case 31: Action that allows a Lead Analyst to edit the properties of an Event

2.2.4. Use Case Scenarios

Below are the interactions that occur between the actors and the actions that can be performed within the F.R.I.C system described in series of steps(Use case scenarios):

Use Case Scenario 1:

Use Case Scenario Name: Manage Task
Description: Action that will allow a lead analyst to identify whether they would like to create, edit, or delete a task
Actors: Lead Analyst, and Database
Pre-condition: The Lead Analyst must be a part of the F.R.I.C system.
Trigger-condition: The Lead Analyst wants to create, edit, or delete a task
Flow of events:

Step 1: System prompts user to create, edit, or delete a task
Step 2: Lead Analyst selects create task
Step 3: System prompts create task page
Step 4: End of use case

Alt Step 2: Lead Analyst selects edit task

Step 2.1: Lead Analyst selects edit task
Step 2.2: System prompts edit task page
Step 2.3: Continue to step 4

Alt Step 2: User selects delete task

Step 2.1: Lead Analyst selects delete task
Step 2.2: System prompts delete task
Step 2.3: Continue to step 4

Use Case Scenario 2:

Use Case Scenario Name: Generate Report

Description: Action that will allow a lead analyst to choose which report they would like to export
Actors: Lead Analyst, Database
Pre-condition: User must be authenticated as a Lead Analyst.
Trigger-condition: Use selects the report they would like to export
Flow of events:

Step 1: Lead Analyst chooses to generate the ERB Report
Step 2: Database retrieves information specific to the report requirements
Step 3: Lead Analyst acknowledges this is the correct information for the report
Step 4: System formats report to match the requirement of the unique report
Step 5: System exports report in pdf format to the Lead Analyst
Step 6: End of use case.

Alt Step 1: Lead Analyst Chooses to generate the Risk Matrix Report

Step 2.1: Use case continues at Step 2

Alt Step 1: Lead Analyst Chooses to generate the Technical Report

Step 2.1: Use case continues at Step 2

Use Case Scenario 3:

Use Case Scenario Name: View progress.
Description: The user (Lead Analyst) would like to see how the progress of the cyber engagement
Actors: Lead Analyst and Database
Pre-condition: User has been verified as a lead analyst
Trigger-condition: The lead analyst requests the current progress of the cyber engagement.
Flow of events:

Step 1: The system prompts the user to determine if they would like the progress of the entire engagement or the progress of an analyst
Step 2: user selects entire engagement
Step 3: The system displays the current progress percentage of combined analysts percentages
Step 4: End Use Case

Alt Step 2: User selects progress of an individual

Step 2.1: The system displays the user information for the individual
Step 2.1: The system displays the current progress percentage for the individual
Step 2.3: Use case continues at Step 4

Use Case Scenario 4:

Use Case Scenario Name: Create Task
Description: Create a brand new task
Actors: Lead Analyst(referred as user) and Database
Pre-condition: The user has to be authenticated as a lead analyst
Trigger-condition: Lead analyst requests to create tasks.

Flow of events:

Step 1: System shows editable fields required for the task (description, etc..)
Step 2: Lead Analyst modifies desired information of the task
Step 3: Lead analyst submits the task information to database
Step 4: System display message to ask for confirmation on the creation of this new task
Step 5: Lead Analyst confirms their action
Step 7: System saves the task into the database
Step 8: End of use case.

Use Case Scenario 5:

Use Case Scenario Name: Edit Task
Description: Edit the properties of a task
Actors: Lead Analyst(referred as user) and Database
Pre-condition: The user has to be authenticated as a lead analyst
Trigger-condition: Lead analyst requests to edit a specific task
Flow of events:

Step 1: System shows information of the selected task
Step 2: User modifies desired information of the task
Step 3: User submits the new task information
Step 4: System display message to ask for confirmation on edited information.
Step 5: User confirms their action
Step 7: System saves the modified task into the database
Step 8: End of use case.

Alt Step 5: The lead analyst doesn't intend to submit the edit.

Step 5.1: System closes the prompt window
Step 5.2: End of use case

Alt Step 7: The system fails to update the record of the finding.

Step 7.1: System alerts the analyst the record was not updated
Step 7.2: System will prompt the user to enter re-enter information
Step 7.3: Continue to step 1

Use Case Scenario 6:

Use Case Scenario Name: Delete task.
Description: Delete a task from the database.
Actors: Lead analyst and Database.
Pre-condition: Lead Analyst must be authenticated in system, the lead analyst must select an already created task to delete.
Trigger-condition: Analyst requests the system to delete a task from the database.
Flow of events:

Step 1: The system asks the lead analyst for a confirmation warning in the deletion of the chosen task.
Step 2: The lead analyst confirms the deletion.

Step 3: The system receives the confirmation to the request.

Step 4: The system deletes the task from the database.

Alt Step 2: The user declines the confirmation.

Step 2.1: End current use case.

Use Case Scenario 7:

Use Case Scenario Name: Create Subtasks.

Description: Create and save a subtask.

Actors: Database, lead analysts and analyst(referred as user).

Pre-condition: A task must first be created or already exist.

Trigger-condition: The user requests to create a new sub tasks while working on a current task.

Flow of events:

Step 1: System display fields required for the subtask.

Step 2: User fills out the fields for a subtask.

Step 3: User presses the save button.

Step 4: System display a message asking for confirmation on the creation of the new subtask

Step 5: User presses the confirm button.

Step 6: System creates record of subtask relating to main tasks on the database.

Step 7: End of use case.

Alt Step 3: User did not fill certain fields.

Step 3.1: System asks the user to fill missing fields.

Step 3.2: User fills missing fields

Step 3.4: User confirms the changes made.

Step 3.5: Use case continue at step 4.

Alt Step 3: User cancels the subtask creation.

Step 3.1: End of use case

Alt Step 4: User declines delete confirmation

Step 4.1: End of use case

Use Case Scenario 8:

Use Case Scenario Name: Edit Subtask.

Description: Change the properties of a subtask.

Actors: Database, lead analyst and analyst(referred as user).

Pre-condition: The analyst must select an already created subtask to delete.

Trigger-condition: The user changes the properties of a certain subtask.

Flow of events:

Step 1: System displays the details and artifacts related to the selected subtask

Step 2: User modifies desired information of the subtask.

Step 3: User confirms the changes made.

Step 4: System display message to ask for confirmation on edited information.
Step 5: User confirms.
Step 6: System saves the modified subtask into the database.
Step 7: End of use case.

Alt Step 3: User did not fill certain fields.
Step 3.1: System asks the user to fill missing fields.
Step 3.2: User fills missing fields
Step 3.3: User confirms the changes made.
Step 3.4: Use case continue at step 3.

Alt Step 3: User cancel the subtask edition.
Step 3.1: End of use case.

Alt Step 5: User declines the confirmation message.
Step 5.1: End of use case.

Use Case Scenario 9:

Use Case Scenario Name: Delete subtask
Description: Delete a subtask from the database.
Actors: Database, lead analyst and analyst(referred as user).
Pre-condition: A subtask must be present and established (already created), and the subtask to be deleted has to be selected.
Trigger-condition: User request to delete a subtask.
Flow of events:
Step 1: System asks the user for confirmation.
Step 2: The user confirms that he wants to delete a subtask.
Step 3: System removes record of subtask from database.
Step 4: End of use case.

Alt Step 2: User declines delete confirmation
Step 2.1: End of use case.

Use Case Scenario 10:

Use Case Scenario Name: Filter data.
Description: Analyst/ Lead Analyst can select what information they want presented to them from tasks, subtasks, and findings
Actors: Lead Analyst, Analyst, and Database
Pre-condition: The data must be present in the database
Trigger-condition: Lead Analyst/Analyst select what they want from the database
Flow of events:
Step 1: System locates desired information for the user.
Step 2: System displays the information based on the needs.
Step 3: Lead Analyst, Analyst can view the information.
Step 4: Use Case Ends.

Alt step 1: Analyst selects data that system cannot find

Step 1: Lead Analyst/ Analyst select an option that doesn't have any information
Step 2: System displays empty display panel.
Step 3: Use Case Ends.

Use Case Scenario 11:

Use Case Scenario Name: Sort data.
Description: Database will organize all task, subtask, and findings for easy finding
Actors: Database
Pre-condition: Information must be present in order for organizing to function properly
Trigger-condition: Information is added to the database
Flow of events:

Step 1: System receives the content
Step 2: System implements algorithm
Step 3: System stores information in data structure
Step 4: Use Case Ends.

Alt step 1: System doesn't have content

Step 1: System content doesn't contain content
Step 2: Use Case Ends.

Use Case Scenario 12:

Use Case Scenario Name: Search data.
Description: Allows user to look through task, sub tasks, and findings
Actors: Lead analyst, analyst, and database
Pre-condition: Must be a valid lookup
Trigger-condition: User searches for something in the database
Flow of events:

Step 1: System views request from user
Step 2: System accesses information
Step 3: System displays the content
Step 4: Lead analyst/ analyst can view information based on the search
Step 5: Use Case Ends.

Alt step 2: System doesn't have content

Step 1: System content doesn't contain content
Step 2: Use Case Ends.

Use Case Scenario 13:

Use Case Scenario Name: Sort tasks.
Description: The system uses the database to sort the tasks according to the lead analyst criteria.
Actors: Lead Analyst and Database
Pre-condition: Tasks have to be made about a system in question.
Trigger-condition: The lead analyst requires the tasks be sorted.

Flow of events:

Step 1: The system prompts the analyst to choose how to sort the tasks.

Step 2: The analyst chooses by priority.

Step 3: The system uses the database to sort the tasks in descending order with highest priority first, the system displays the tasks.

Step 4: The analyst chooses to sort the task by progress.

Step 5: The system uses the database to sort the tasks in descending order with highest progress first, the system displays the tasks.

Step 6: The analyst chooses sort alphabetically.

Step 7: The system uses the database to sort the tasks in alphabetic order, the system displays the tasks.

Step 7: The analyst chooses to sort by date.

Step 8: The system uses the database to sort the tasks in descending order with the oldest date first, the system displays the tasks.

Step 9: End of use case.

Alt step 1: Sort algorithm isn't prompted

Step 1: System error

Step 2: Continue to step 1

Use Case Scenario 14:

Use Case Scenario Name: Sort Findings

Description: Arrange the findings in a way specified by the analyst and/or lead analyst

Actors: Lead analyst, analyst (referred to as users), and Database

Pre-condition: Findings need to exist to sort

Trigger-condition: User requests data to be sorted

Flow of events:

Step 1: User specifies how the findings are to be sorted based on the attribute(e.g. title , due date,priority, etc...) of the finding.

Step 2: System reads the findings from the database and arranges according to the specified attribute.

Step 3: End of use case .

Alt Step 2: The system fails to retrieve findings.

Step 2.1: System alerts the analyst no findings are found

Step 2.2: System will prompt the user to enter re-specify the attribute

Step 2.3: Continue to step 1

Use Case Scenario 15:

Use Case Scenario Name: Search Task.

Description: Action that allows the user to look for a certain task.

Actors: Lead analyst, Analyst (referred to as users), and Database.

Pre-condition: Database must have content inorder to look up the task.

Trigger-condition: The user prompts a search for a certain task.

Flow of events:

Step 1: The database receives the request.

Step 2: The database sends back the information requested.

Step 3: End of use case.

Alt Step 1: Cannot find the data requested by the analyst or lead analyst.

Step 1.1: The user is prompted with an error message.

Step 1.2: End current use case.

Use Case Scenario 16:

Use Case Scenario Name: Search Findings.

Description: Allows the user to look for a certain finding.

Actors: Lead Analyst, Analyst, and Database.

Pre-condition: Database must have content inorder to look up a finding.

Trigger-condition: The user prompts a search for a certain finding..

Flow of events:

Step 1: The database receives the request.

Step 2: The database sends back the information requested.

Step 3: End of use case.

Alt Step 1: Cannot find the data requested by the analyst or lead analyst.

Step 1.1: The user is prompted with an error message.

Step 1.2: End current use case.

Use Case Scenario 17:

Use Case Scenario Name: Sync Data.

Description: Will compare current data of the analyst and lead analyst to compile the most recent information.

Actors: Lead Analyst, Analyst (referred to as users) and Database.

Pre-condition: The analyst and lead analyst has to be authenticated with the database.

Trigger-condition: Lead analyst requests data to be synced.

Flow of events:

Step 1: System check who the user is currently assigned to

Step 2: System obtains all current user data for assigned team from database.

Step 3: System compares who has the most current edit.

Step 4: System compiles data.

Step 5: System saves information to database.

Step 5: System displays organized data.

Step 6: End current use case.

Alt Step 2: **Cannot sync with an analyst**

Step 2.1: System cannot sync data from an user.

Step 2.2: Show warning message that it wasn't able to obtain data from # analyst.

Step 2.3: Show which analyst the DB was not able to get from.

Use Case Scenario 18:

Use Case Scenario Name: Manage Finding.
Description: View the findings available to the analyst(s) with the purpose of viewing, editing, or deleting a finding.
Actors: Lead analyst, analyst, and database
Pre-condition: Analyst/Lead Analyst have to be part of the F.R.I.C system.
Trigger-condition: Analyst/Lead Analyst want to create, edit, or delete a finding
Flow of events:

Step 1: System prompts user to create, edit, or delete a findings
Step 2: User selects create findings
Step 3: System prompts create findings
Step 4: End of use case

Alt Step 2: User selects edit findings

Step 2.1: User selects edit findings
Step 2.2: System prompts edit findings
Step 2.3: Continue to step 4

Alt Step 2: User selects delete findings

Step 2.1: User selects delete findings
Step 2.2: System prompts delete findings
Step 2.3: Continue to step 4

Use Case Scenario 19:

Use Case Scenario Name: Create finding
Description: Create a record of information found during an analysis
Actors: Lead analyst, Analyst (referred to as users), and Database
Pre-condition: A user has a finding they wish to enter to F.R.I.C
Trigger-condition: User requests to enter a finding
Flow of events:

Step 1: System prompts the analyst to enter information related to the finding
Step 2: User enters information of the finding
Step 3: System uses the database to create a record of the finding.
Step 4: System prompts the user to enter evidence of the finding. (Screenshots,Video, a file etc…)
Step 5: User enters the evidence needed for the finding
Step 6: The system uses the database to add the provided information of the finding.
Step 7: End of use case.

Alt Step 3: The system fails to enter information to the database.

Step 3.1: System will prompt the user to try submitting again
Step 3.2: End of use case.

Use Case Scenario 20:

Use Case Scenario Name: Edit finding

Description: Action that allows the user to change the properties in a certain finding
Actors: Lead analyst, Analyst (referred to as users), and Database
Pre-condition: A finding exists to be edited
Trigger-condition: User requests to modify a specific finding
Flow of events:

Step 1: System displays the details and artifacts related to the selected finding
Step 2: User will modify the desired properties of the finding
Step 3: User will submit the modified version of the finding
Step 4: System will prompt the user to verify their intention to modify a finding
Step 5: User will confirm their action
Step 6: System will use the database to update the finding record
Step 7: System will confirm the record has been updated
Step 8: End of use case

Alt Step 6: The system fails to update the record of the finding.

Step 6.1: System alerts the analyst the record was not updated
Step 6.2: System will prompt the user to enter re-enter information
Step 6.3: Continue to step 1

Use Case Scenario 21:

Use Case Scenario Name: Delete finding
Description: An analyst and/or lead analyst wants to delete a finding
Actors: Lead analyst, Analyst (referred to as users), and Database
Pre-condition: A user has selected a finding to delete
Trigger-condition: The user requests to delete a finding
Flow of events:

Step 1: The system prompts the user with a warning message to confirm the deletion of the finding.
Step 2: The lead analyst confirms the deletion.
Step 3: The system receives the confirmation to the request
Step 4: The system deletes the findings from the database

Alt Step 2: The user declines the confirmation:

Step 2.1: System will close the warning dialog
Step2.1: End of use case

Use Case Scenario 22:

Use Case Scenario Name: Login In
Description: An analyst and lead analyst will use their credentials to login.
Actors: Database, Lead Analyst, & Analyst (Analyst and Lead Analyst will be referred to as user)
Pre-condition: The user must have created an account with F.R.I.C.
Trigger-condition: A user opens the F.R.I.C application
Flow of events:

Step 1: System asks for the users initials as their credentials
Step 2: User enters their credentials
Step 3: System uses the database to verify the entered information
Step 4: System confirms the users credentials
Step 5: User is granted access to F.R.I.C.
Step 4: End of use case.

Alt Step 4: The system does not confirm the user credentials:
Step 4.1: System denies access
Step 4.2: System prompts the user to enter their credentials again
Step 4.3: Continues to step 2

Use Case Scenario 23:

Use Case Scenario Name: Push Data.
Description: System allows any analyst to upload their current data to the main system.
Actors: Lead Analyst, Analyst, Database.
Pre-condition: The analyst and lead analyst has to be authenticated with the database.
Trigger-condition: Analyst requests data to be pushed.
Flow of events:
Step 1: System confirms that their current data will be pushed to the database.
Step 2: System obtains the analyst current data to be pushed.
Step 3: System merges with the previous data from the database.
Step 4: System updates the main data with the new changes.
Step 5: System makes new updated data pullable to all other analysts.
Step 7: System updates timestamp.
Step 8: End of use case

Alt Step 1: System confirmed that their data will be pushed to a specific user.
Step 1.1: System confirms that their current data will be pushed to a specific analyst.
Step 1.2: System obtains the analyst current data to be pushed.
Step 1.3: System merges with the previous data from the analyst.
Step 1.4: System updates the analyst data with the new changes.
Step 1.5: System updates timestamp.
Step 1.6: End of use case

Use Case Scenario 24:

Use Case Scenario Name: Pull Data
Description: System allows any analyst to receive data that has been pushed by another analyst.
Actors: Lead Analyst, Analyst, Database.
Pre-condition: Analyst/Lead Analyst have to be registered in the F.R.I.C system.
Trigger-condition: Any analyst selects the sync feature
Flow of events:
Step 1: System displays changes between analyst and database.

Step 2: System confirms data to be pulled.
Step 3: System obtains the main data from the database to be pulled.
Step 4: System merges the previous data from the analyst with the pulled data.
Step 5: System updates the local data with new changes to the database.
Step 6: System updates timestamp.
Step 7: End of use case.

Alt Step 1: System confirms pulling data from a specific user.

Step 1.1: System displays changes between analyst and other analyst data.
Step 1.2: System confirms data to be pulled.
Step 1.3: System obtains the main data to be pulled.
Step 1.4: System merges the previous data from the analyst with the other analyst data.
Step 1.5: System updates the local data with new changes.
Step 1.6: System updates timestamp.
Step 1.7: End of use case.

Use Case Scenario 25:

Use Case Scenario Name: Create Event
Description: Create a brand new Event
Actors: Lead Analyst(referred as user) and Database
Pre-condition: The user has to be authenticated as a lead analyst
Trigger-condition: Lead analyst requests to create an Event
Flow of events:

Step 1: System shows editable fields required for the Event (description, etc..)
Step 2: Lead Analyst modifies desired information of the Event
Step 3: Lead analyst submits the Event information to database
Step 4: System display message to ask for confirmation on the creation of this new Event
Step 5: Lead Analyst confirms the creation of the event
Step 7: System saves the task into the database
Step 8: End of use case.

Use Case Scenario 26:

Use Case Scenario Name: Generate ERB
Description: Action that will allow the Lead Analyst to generate a ERB report
Actors: Lead Analyst, Database
Pre-condition: User must be authenticated as a Lead Analyst.
Trigger-condition: User selects the ERB report to export
Flow of events:

Step 1: Database retrieves information specific to the report requirements
Step 2: Analyst acknowledges this is the correct information for the report
Step 3: System formats report to match the requirement of the unique report
Step 4: System exports report in powerpoint format to the Lead Analyst

Step 5: End of use case.

Use Case Scenario 27:

Use Case Scenario Name: Generate Technical Report
Description: Action that will allow to generate a Technical Reportreport
Actors: Lead Analyst, Database
Pre-condition: User must be authenticated as a Lead Analyst.
Trigger-condition: User selects the Technical Report to export
Flow of events:

Step 1: Database retrieves information specific to the report requirements
Step 2: Analyst acknowledges this is the correct information for the report
Step 3: System formats report to match the requirement of the unique report
Step 4: System exports report in Word format to the Lead Analyst
Step 5: End of use case.

Use Case Scenario 28:

Use Case Scenario Name: Generate Risk Matrix
Description: Action that will allow to generate a Risk Matrix report
Actors: Lead Analyst, Database
Pre-condition: User must be authenticated as a Lead Analyst.
Trigger-condition: User selects the Risk Matrix report to export
Flow of events:

Step 1: Database retrieves information specific to the report requirements
Step 2: Analyst acknowledges this is the correct information for the report
Step 3: System formats report to match the requirement of the unique report
Step 4: System exports report in excel format to the Lead Analyst
Step 5: End of use case.

Use Case Scenario 29:

Use Case Scenario Name: Manage Event
Description: Lead analyst(s) request to add or change event information
Actors: Lead Analyst, and Database
Pre-condition: The Lead Analyst must be a part of the F.R.I.C system.
Trigger-condition: The Lead Analyst wants to create, edit, or delete an event
Flow of events:

Step 1: System prompts user to create, edit, or delete an event
Step 2: Lead Analyst selects create event
Step 3: System prompts create event
Step 4: End of use case

Alt Step 2: Lead Analyst selects edit event

Step 2.1: Lead Analyst selects edit event

Step 2.2: System prompts edit event
Step 2.3: Continue to step 4

Alt Step 2: User selects delete event
Step 2.1: Lead Analyst selects delete event
Step 2.2: System prompts delete event
Step 2.3: Continue to step 4

Use Case Scenario 30:

Use Case Scenario Name: Delete Event
Description: Delete an established Event from the database
Actors: Lead analyst and Database.
Pre-condition: Lead Analyst must be authenticated in system, the lead analyst must select an already created Event to delete.
Trigger-condition: Analyst sends request to the system to delete an Event from the database.
Flow of events:
Step 1: The system prompts the user with a warning message to confirm the deletion of the Event
Step 2: The lead analyst confirms the deletion.
Step 3: The system receives the confirmation to the request.
Step 4: The system requests the Leads Analyst Initials
Step 5: The system makes note of who deleted the Event and send to database
Step 5: The system deletes the Event from the database

Alt Step 2: The user declines the confirmation.
Step 2.1: End current use case

Use Case Scenario 31:

Use Case Scenario Name: Edit Event
Description: Action that allows a Lead Analyst to edit the properties of an Event
Actors: Lead Analyst and Database
Pre-condition: The user has to be authenticated as a lead analyst
Trigger-condition: Lead analyst requests to edit a specific Event
Flow of events:
Step 1: Lead Analyst selects a specific Event
Step 1: System shows information of the selected task
Step 2: Lead Analyst modifies desired information of the Event
Step 3: Lead Analyst submits the new Event information
Step 4: System prompts user to ask for confirmation on edited information.
Step 5: User confirms their action
Step 7: System saves the modified Event and send to the database
Step 8: End of use case.

Alt Step 5: The lead analyst doesn't intend to submit the edit.
Step 5.1: System closes the prompt window

Step 5.2: End of use case

2.3. User Characteristics

This section will define the different users that interact with F.R.I.C. It will also describe the purpose of the interactions as well as the difference between the different types of users and their purpose.

There are two types of users that interact with the system: lead analyst and analyst. They have similar uses of the system except the lead analyst has a wider range of access to other analysts' work.

Analysts use F.R.I.C to complete tasks that have been assigned by the lead analyst or they can pick up tasks at the discretion of the lead analyst. Analysts can create and archive findings and subtasks that they create. They also have the ability to add another analyst as a collaborator giving the other analyst edit permissions only. Analysts have the ability to edit task status and enter information related to the task. All information entered is available for analysts to see, the artifacts that are displayed on the system can only be editable by the creator and the lead analyst.

Lead analysts use F.R.I.C to manage the other analysts and the current assessment event. The lead analyst has the ability to create an event and enter systems and tasks that can either be assigned or analysts can be given the freedom to pick up a task. A lead analyst has the freedom to view and edit all findings,tasks, and subtasks.

2.4. General Constraints

The following section includes known constraints of the F.R.I.C system during implementation. Listed below are the constraints given by the clients based on security or other aspects that weren’t mentioned.

Constraint 1: The system shall be only accessible to the users who meet the credentials
Constraint 2: The system shall be on a private server no internet connection
Constraint 3: The duration of the system shall be completed by the end of 2020 Fall semester

2.5. Assumptions and Dependencies

TIn order to provide a more clear description of the requirements of F.R.I.C the development team has made the following assumptions and dependencies about F.R.I.C. below:

2.5.1. Assumptions:

- The system shall run on a Kali Linux operating system.
- The source code shall be modifiable after the system has been delivered to the clients (maintenance).
- The clients shall have the necessary hardware to run the F.R.I.C system.
- Development of the application will be done in Python.
- C.E.A.D will provide a template for the reports.
- F.R.I.C shall be available as a web application.

2.5.2. Dependencies:

- The components of the system are based on the SRS, if any changes are made it will affect the F.R.I.C system.

3. Specific Requirements

3.1. External Interface Requirements

These sections seek to convey the requirements of the F.R.I.C. system included in: user, hardware, software, and communication interfaces. Listed in the following sections included requirements and their different components needed to satisfy the clients needs.

3.1.1. User Interfaces

This section illustrates requirement components needed in the prototype in order to make the F.R.I.C. system. Listed below are the following requirements and the prototype associated with them

[SRS 1] The following Sign in page shall contain the following components in Figure 1.1. The system shall allow any analyst to log in with their initials. The system shall allow two different types of users to login to F.R.I.C, a Lead Analyst and an Analyst.

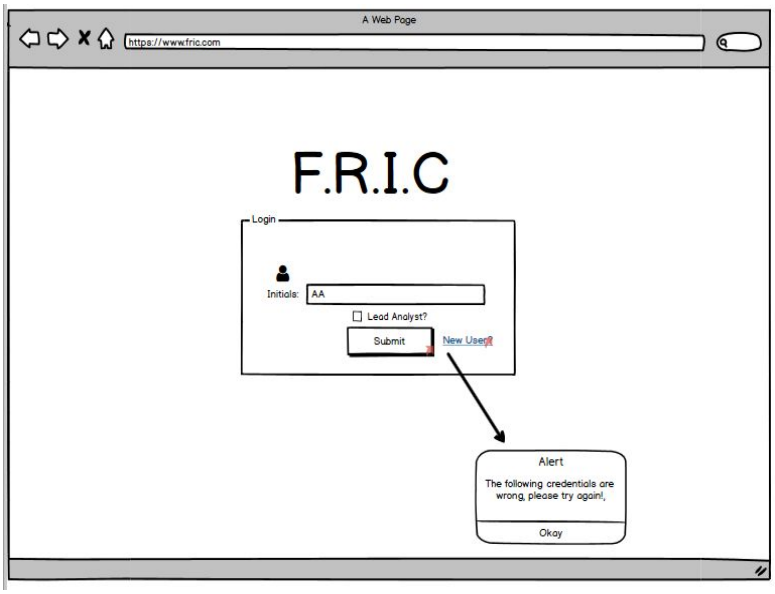


Figure 1.1: Sign in page.

A. Text box labeled as “Initials”.
B. Checkbox labeled as “Lead analyst”.
C. Button labeled as “Submit”.
D. Hyperlink labeled as “New Users?” .

[SRS 2] The following Sign up page shall contain the following components in Figure 2.1. The system shall allow any analyst to create an account with their initials.

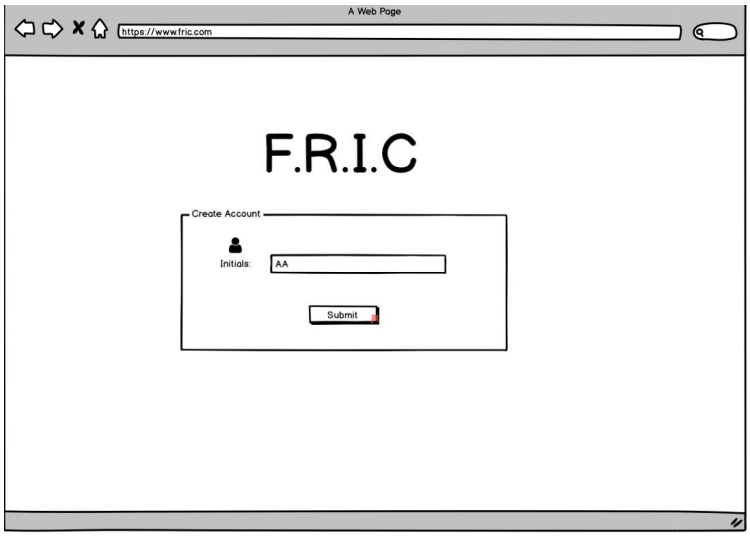


Figure 2.1: Sign up page.

A. Text box labeled as “Initials”.
B. Button labeled as “Submit”.

[SRS 3] The system shall have an event homepage that contains the following components shown in figure 3.1. The event homepage has all the controls a lead analyst needs to manage events and systems. While providing key information to analysts.

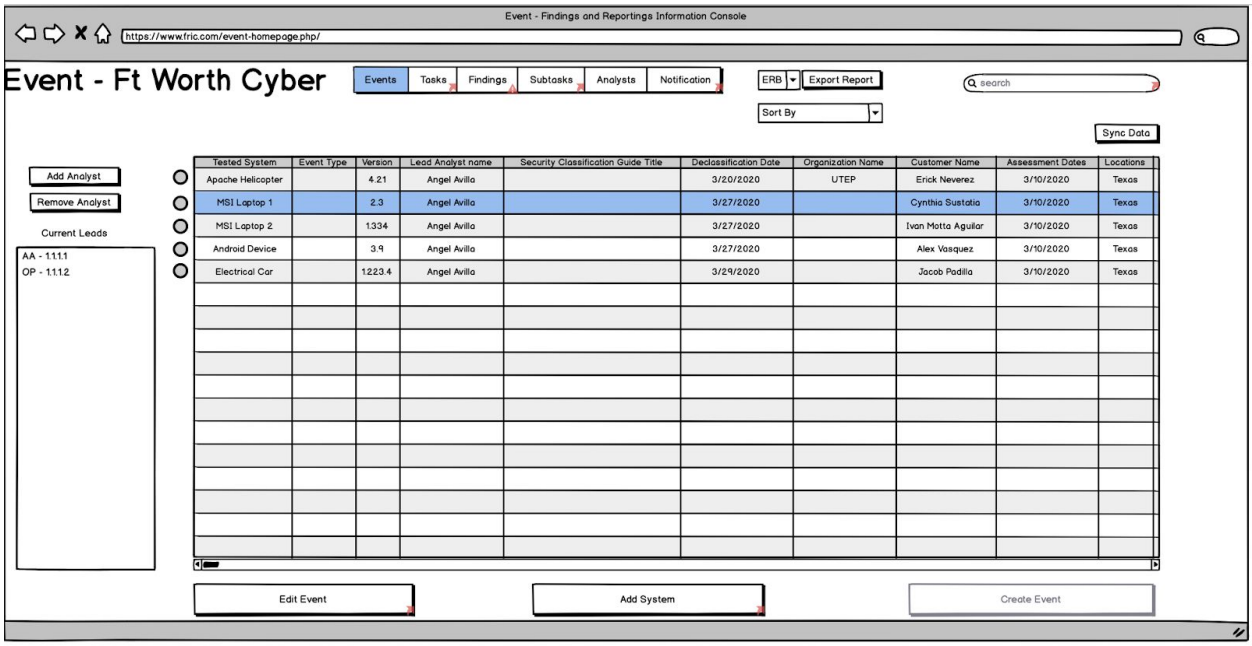


Figure 3.1: Event Homepage

A. Text label that displays the Event name.
B. Navigation bar to navigate to other views through the use of buttons including:
    a. Events.
    b. Tasks.
    c. Findings.
    d. Subtasks.
    e. Analyst.
    f. Notifications.
C. “Report” Drop down menu to choose a report to export to.
D. Button to allow a report to be exported.
E. “Sort By” Drop down menu that changes sort criteria.
F. Search bar to look for systems.
G. Button to sync data.
H. Table that contains the systems of an event with the columns:
    a. Tested system name.
    b. Event type.
    c. Version.

d. Security classification guide title.
e. Declassification Date.
f. Organization name.
g. Customer name.
h. Assessment dates.
i. Locations.

I. Check bubble that allows certain systems to be selected for exporting to a report.
J. Button to add an analyst .
K. Button to remove an analyst.
L. A box view that contains a list of the lead analysts for the event.
M. Button Edit event to change event attributes.
N. Button Add System to add a system to the table.
O. The components are wrapped in a web browser view to indicate a web based application.

[SRS 4] The system shall have a findings page with the following components shown in figure 4.1: The findings homepage has all the controls a lead analyst needs to manage findings. While providing key information to analysts.

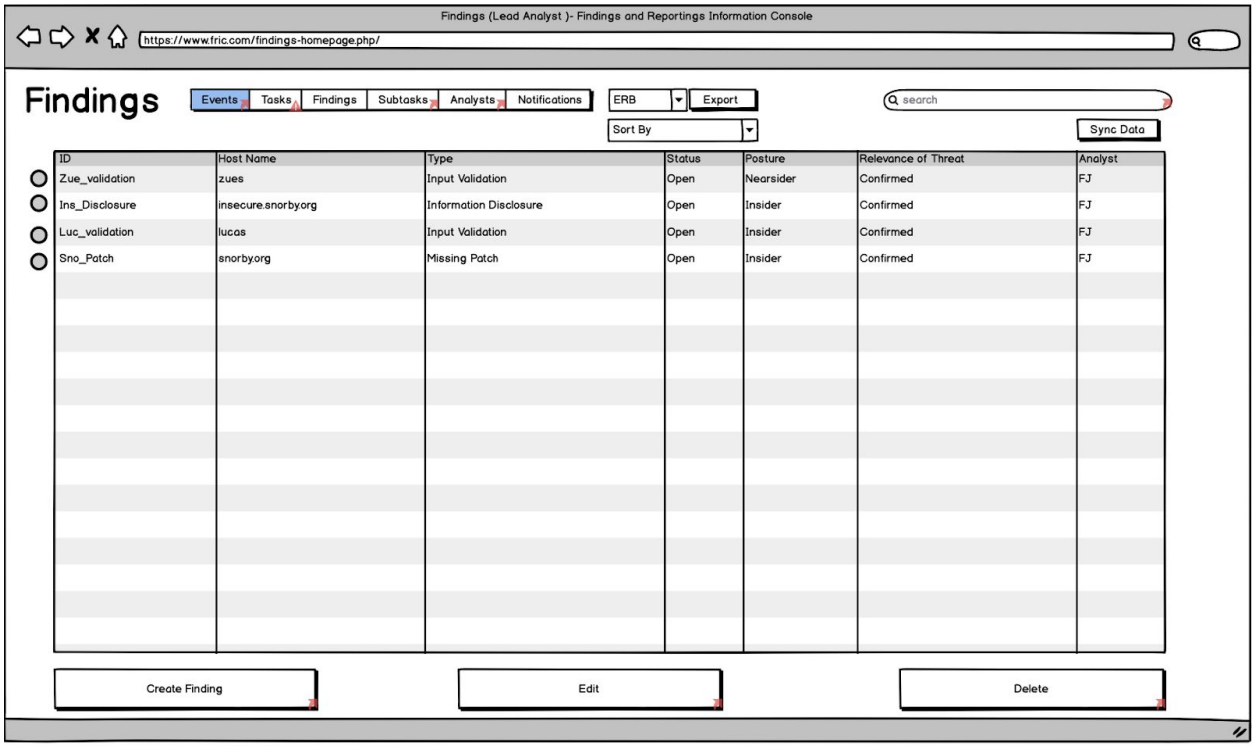


Figure 4.1 Findings Page

A. Text label that displays the page name.
B. Toolbar to navigate to other views.
C. Drop down menu to choose a report to export to.
D. Button to allow a report to be exported.

E. Drop down menu that changes sort criteria.
F. Search bar to look for systems.
G. Button to sync data.
H. Check bubble that allows certain systems to be selected for exporting to a report.
I. Table that contains a list of findings.
    a. ID.
    b. Host name.
    c. Type.
    d. Status.
    e. Posture.
    f. Relevance of Threat.
    g. Analyst.
J. Button to create a new finding.
K. Button to edit a selected finding.
L. Button to delete a selected finding.
M. The components are wrapped in a web browser view to indicate a web based application.

[SRS 5] The system shall have a view to choose which analyst receives updated information which contains the following components shown in figure 5.1.
Text label to indicate the following list is to choose an analyst(s).Table to display a list of analysts.Table to display a list of analysts

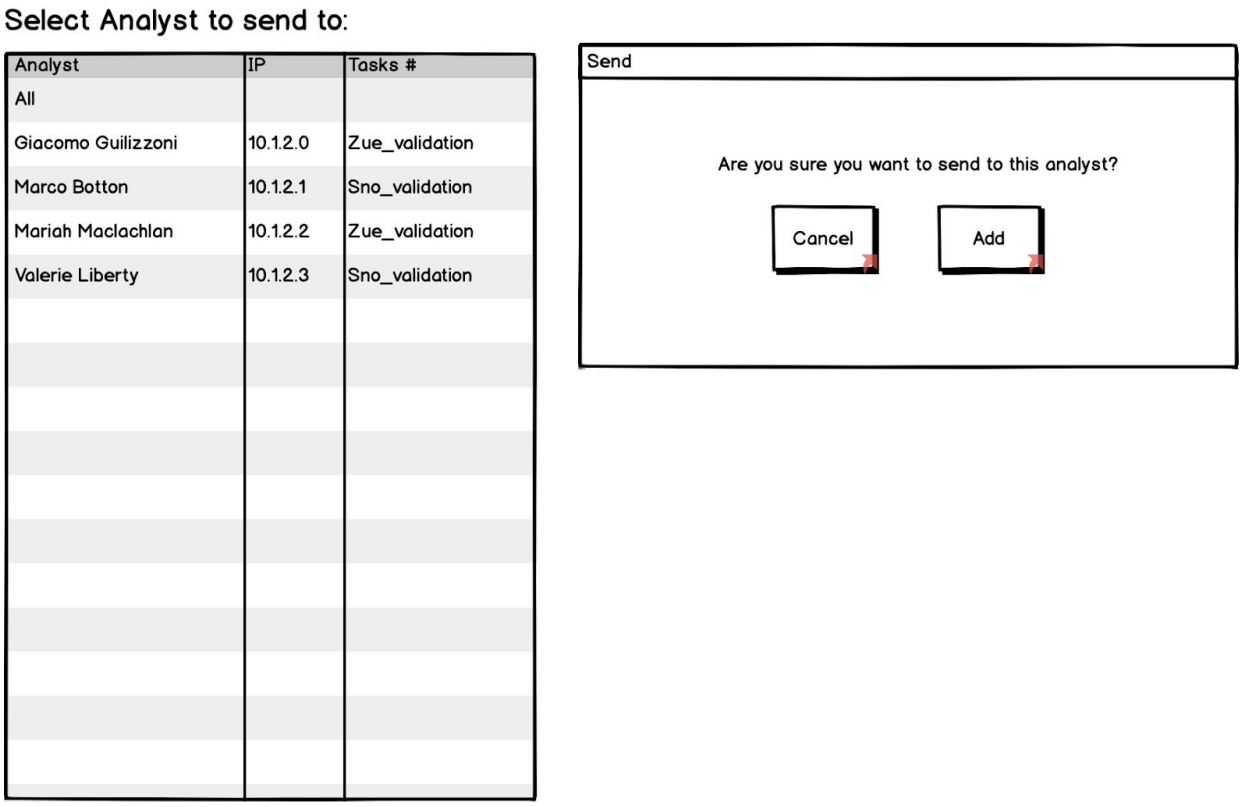


Figure 5.1:Sync View.

a. Analysts Initials.
b. Analysts IP.
c. Analysts task count.

[SRS 6] The system shall have an alert box view that verifies user action that includes the components shown in figure 6.1:
This view is to verify user actions.

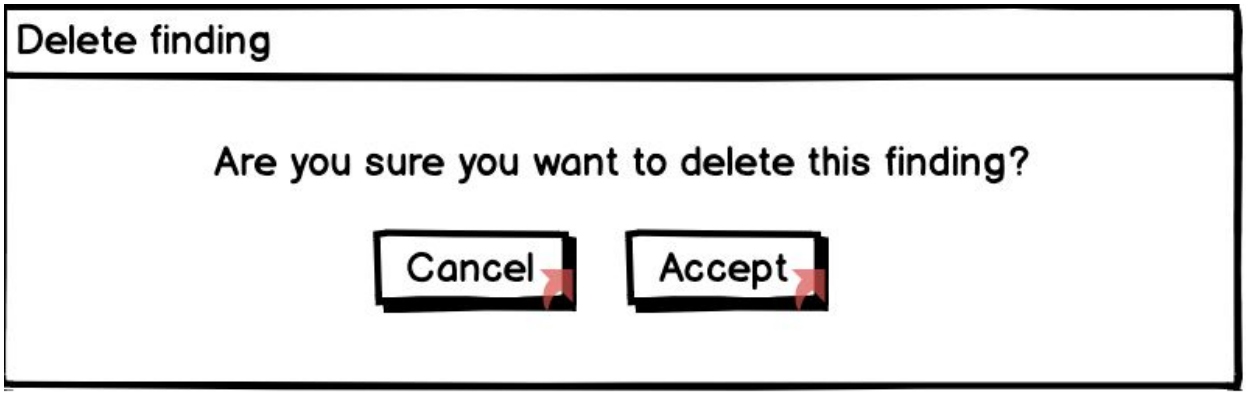


Figure 6.1: Delete Verification

A. Button to cancel action.
B. Button to confirm sync.

[SRS 7] The following Create finding page shall contain the following components in Figure 7.1. The system shall provide multiple fields of text boxes for storing key information for an event.

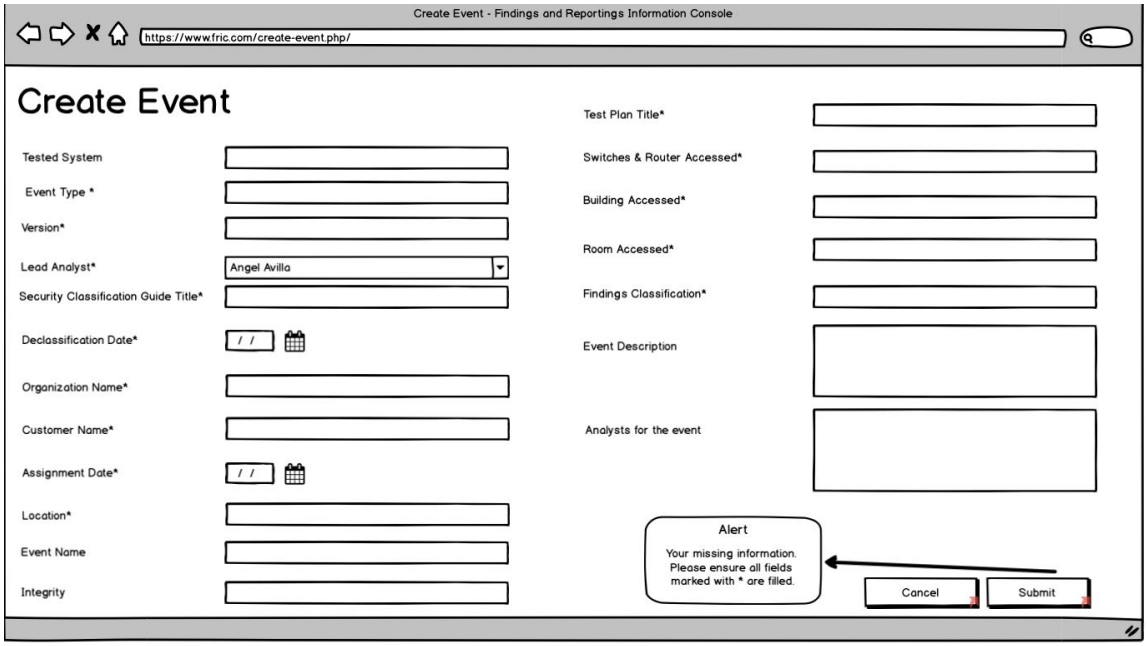


Figure 7.1: Create event page

A. Text box labeled as “Tested System”.
B. Text box labeled as “Event Type”.
C. Text box labeled as “Version”.
D. Dropdown menu labeled as “Lead Analyst”
E. Text box labeled as “Security Classification Guide Title”.
F. Text box labeled as “Declassification Date”.
G. Text box labeled as “Organization Name”.
H. Text box labeled as “Customer Name”.
I. Text box labeled as “Assignment Date”.
J. Text box labeled as “Location”.
K. Text box labeled as “Event Name”.
L. Text box labeled as “Integrity”.
M. Text box labeled as “Test Plan Title”.
N. Text box labeled as “Switches & Routers Accessed”.
O. Text box labeled as “Building Accessed”.
P. Text box labeled as “Room Accessed”.
Q. Text box labeled as “Finding Classification”.
R. Text box labeled as “Event description”.

S. Text box labeled as “Analyst for the event”.
T. Button labeled as “Cancel”.
U. Button labeled as “Submit”.

[SRS 8] The following Create finding page shall contain the following components in Figure 8.1. The system shall provide multiple fields of text boxes for editing information for an event.

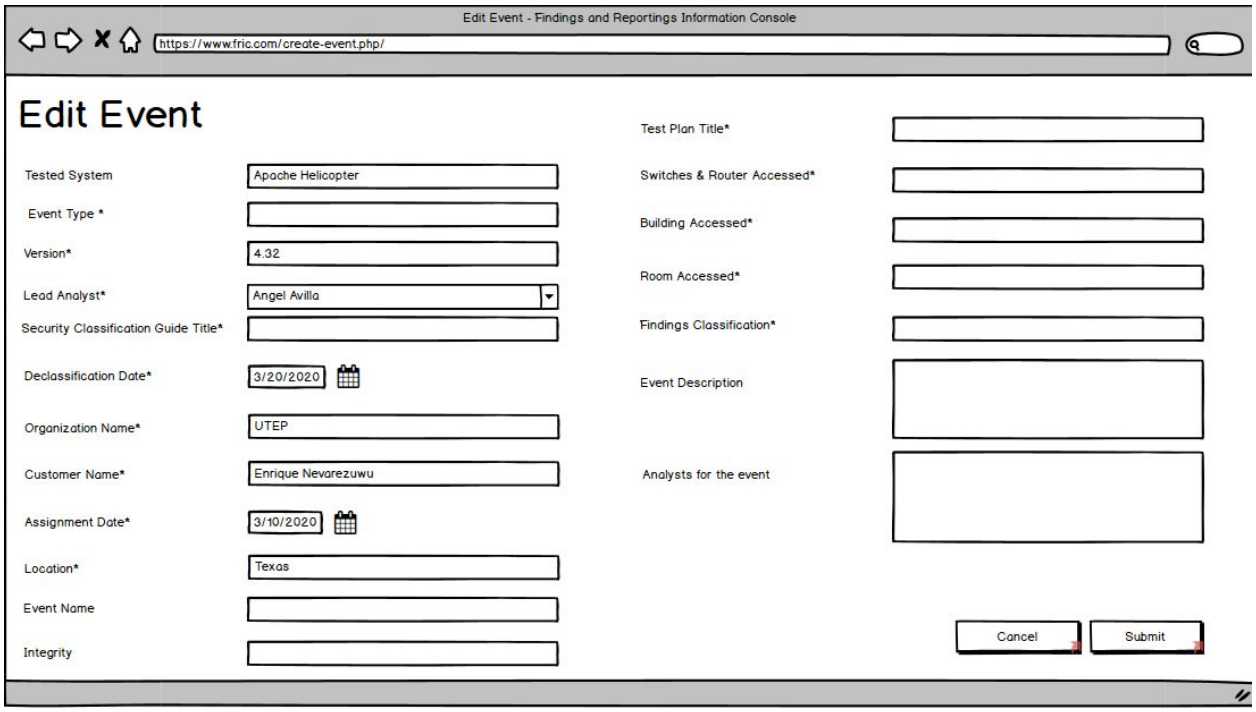


Figure 8.1: Edit event page.

A. Text box labeled as “Tested System”.
B. Text box labeled as “Event Type”.
C. Text box labeled as “Version”.
D. Dropdown menu labeled as “Lead Analyst”.
E. Text box labeled as “Security Classification Guide Title”.
F. Text box labeled as “Declassification Date”.
G. Text box labeled as “Organization Name”.
H. Text box labeled as “Customer Name”.
I. Text box labeled as “Assignment Date”.
J. Text box labeled as “Location”.
K. Text box labeled as “Event Name”.
L. Text box labeled as “Integrity”.
M. Text box labeled as “Test Plan Title”.
N. Text box labeled as “Switches & Routers Accessed”.
O. Text box labeled as “Building Accessed”.

P. Text box labeled as “Room Accessed”.
Q. Text box labeled as “Finding Classification”.
R. Text box labeled as “Event description”.
S. Text box labeled as “Analyst for the event”.
V. Button labeled as “Cancel”
W. Button labeled as “Submit”

[SRS 9]The following Create finding page shall contain the following components in Figure 9.1. The system shall provide multiple fields of text boxes for storing key information for a system. The system shall provide multiple fields of text boxes for storing key information for a system.

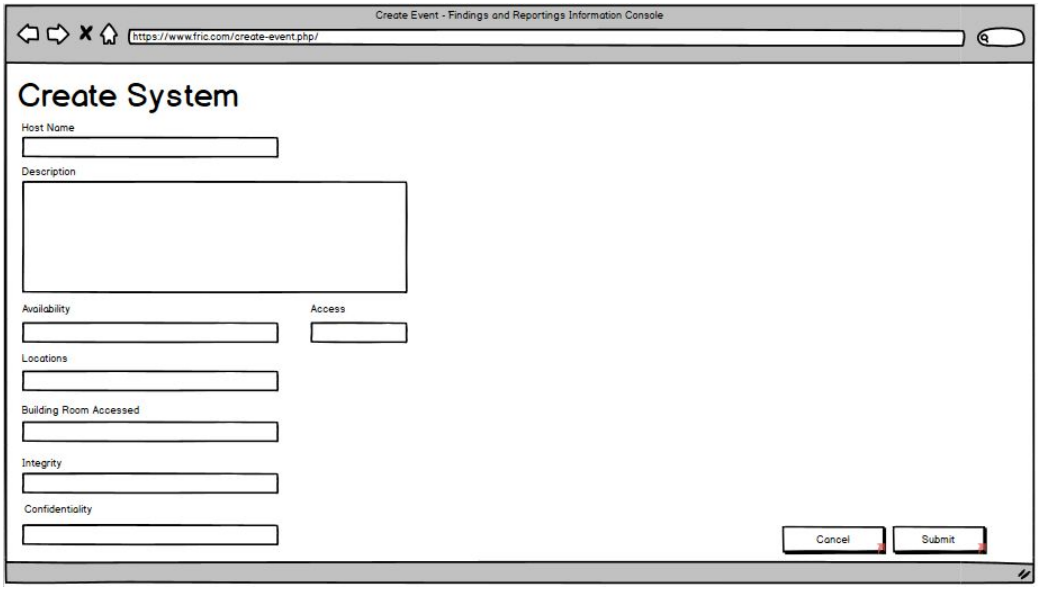


Figure 9.1: Create system page.

A. Text box labeled as “Host Name”.
B. Text box labeled as “Description”.
C. Text box labeled as “Availability”.
D. Text box labeled as “Access”.
E. Text box labeled as “Locations”.
F. Text box labeled as “Building Room Accessed”.
G. Text box labeled as “Integrity”.
H. Text box labeled as “Confidentiality”.
I. Button labeled as “Cancel”.
J. Button labeled as “Submit”.

[SRS 10] The system shall display a Create Task page which contains the following components as shown in Figure 10.1. The system shall allow a lead Analyst to create a task and input the following information.

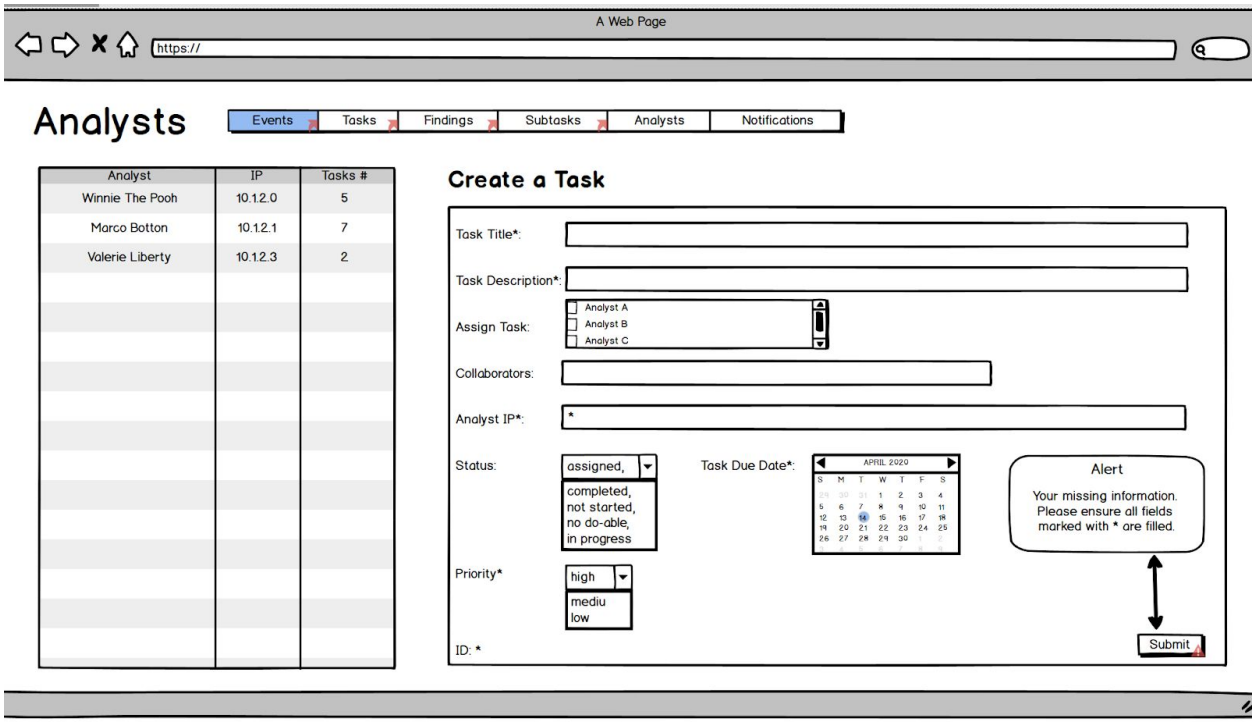


Figure 10.1 Create Task

A. Overlay title labeled as “Analyst”.
B. Navigation bar to navigate to other views through the use of buttons including:
    a. Events.
    b. Tasks.
    c. Findings.
    d. Subtasks.
    e. Analysts.
    f. Notifications.
C. A table which displays the following contents:
    a. Analyst Name.
    b. Analyst IP.
    c. Analyst Tasks Number.
D. A text title named “Create a Task”.
E. A text box named “Task Title”.
F. A text box named “Task Description”.
G. A checklist with analyst names named “Assign Task”.
H. A text box named “Collaborators”.
I. A text box named “Analyst IP”.

J. A drop down button named “Status”.
K. A drop down button named “Priority”.
L. A text field named “ID”.
M. A calendar labeled “Task Due Date” .
N. A button labeled “Submit”.
A. A report drop down menu to choose which report to export to.
B. Export button that exports the tasks to a report.

[SRS 11] The following Create finding page shall contain the following components in Figure 11.1. The system shall provide multiple fields of text boxes for storing key information for a finding.

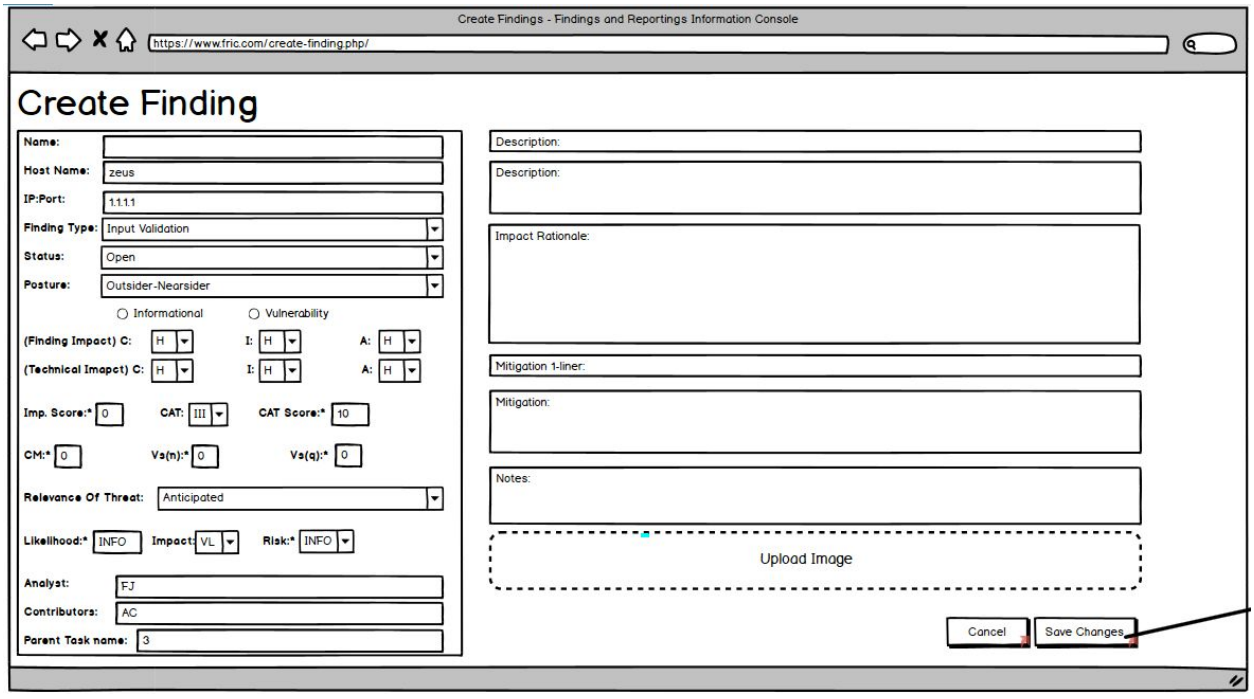


Figure 11.1: Create finding page.

A. Text box labeled as “Name:”.
B. Text box labeled as “Host Name:”.
C. Text box labeled as “IP:Port:”.
D. Dropdown menu labeled as “Finding Type”.
E. Dropdown menu labeled as “Status”.
F. Dropdown menu labeled as “Posture”
G. Radio button labeled as “Informational”.
H. Radio button labeled as “Vulnerability”.
I. Drop down menu labeled as “C”.
J. Drop down menu labeled as “I”.
K. Drop down menu labeled as “A”.

L. Drop down menu labeled as “C”.
M. Drop down menu labeled as “I”.
N. Drop down menu labeled as “A”.
O. Drop down menu labeled as “Imp. Score:”.
P. Text box labeled as “CAT:”.
Q. Text box labeled as “CAT SCORE:”.
R. Text box labeled as “CM:”.
S. Text box labeled as “Vs(n):”.
T. Text box labeled as “Vs(q):”.
U. Drop down menu labeled as “Relevance Of Threat:”.
V. Text box labeled as “Likelihood:”.
W. Drop down menu labeled as “Impact:”.
X. Drop down menu labeled as “Risk:”.
Y. Text box labeled as “Analyst:”.
Z. Text box labeled as “Contributors:”.
AA. Text box labeled as “Parent Task name:”.
BB. Text box labeled as “Short Description:”.
CC. Text box labeled as “Description:”.
DD. Text box labeled as “Impact Rationale:”.
EE. Text box labeled as “Mitigation 1-liner:”.
FF. Text box labeled as “Mitigation”.
GG. Text box labeled as “Notes”.
HH. Drag and drop
II. Button labeled as “Cancel”
JJ. Button labeled as “Save changes”

[SRS 12] The following Edit finding page shall contain the following components in Figure 12.2. The system shall provide multiple fields of text boxes for editing information for a finding.

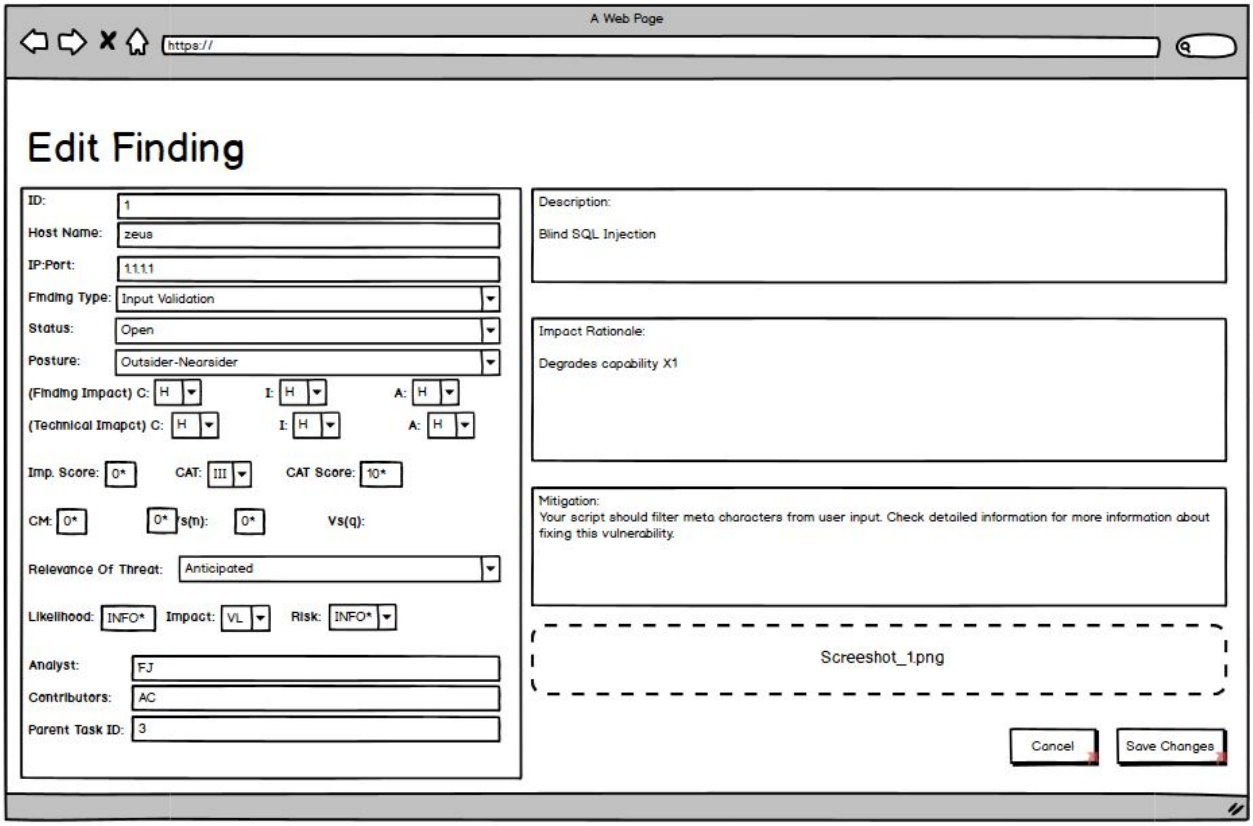


Figure 12.1:Edit finding page.

The system shall provide multiple fields of text boxes for storing key information for a finding.

A. Text box labeled as “Name:”.
B. Text box labeled as “Host Name:”.
C. Text box labeled as “IP:Port:”.
D. Dropdown menu labeled as “Finding Type”.
E. Dropdown menu labeled as “Status”.
F. Dropdown menu labeled as “Posture”
G. Radio button labeled as “Informational”.
H. Radio button labeled as “Vulnerability”.
I. Drop down menu labeled as “C”.
J. Drop down menu labeled as “I”.
K. Drop down menu labeled as “A”.
L. Drop down menu labeled as “C”.
M. Drop down menu labeled as “I”.
N. Drop down menu labeled as “A”.
O. Drop down menu labeled as “Imp. Score:”.
P. Text box labeled as “CAT:”.
Q. Text box labeled as “CAT SCORE:”.
R. Text box labeled as “CM:”.
S. Text box labeled as “Vs(n):”.

T. Text box labeled as “Vs(q):”.
U. Drop down menu labeled as “Relevance Of Threat:”.
V. Text box labeled as “Likelihood:”.
W. Drop down menu labeled as “Impact:”.
X. Drop down menu labeled as “Risk:”.
Y. Text box labeled as “Analyst:”.
Z. Text box labeled as “Contributors:”.
AA. Text box labeled as “Parent Task name:”.
BB. Text box labeled as “Short Description:”.
CC. Text box labeled as “Description:”.
DD. Text box labeled as “Impact Rationale:”.
EE. Text box labeled as “Mitigation 1-liner:”.
FF. Text box labeled as “Mitigation”.
GG. Text box labeled as “Notes”.
HH. Drag and drop section.
II. Button labeled as “Cancel”.
JJ. Button labeled as “Save changes”.

[SRS 13] The following Finding Information should contain the following components in Figure 13.1

The system shall display an overlay that will show the analyst the current info from the selected Findings.

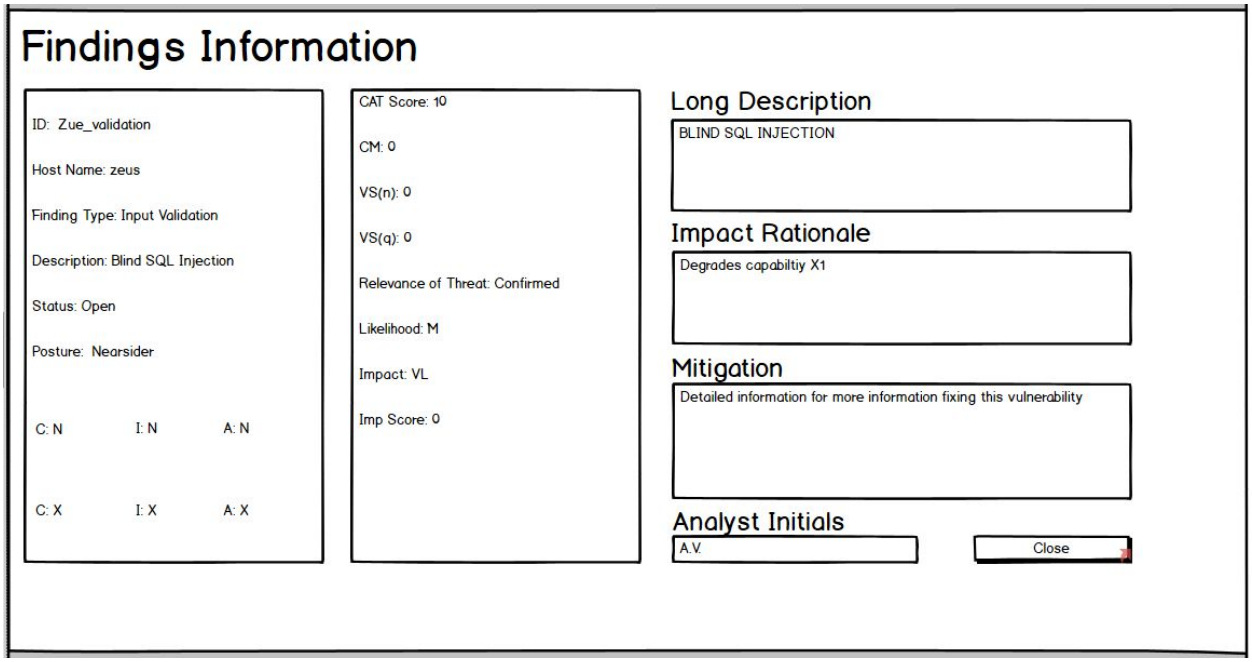


FIgure 13.1:Findings Information.

A. Text field labeled as “ID”.
B. Text field labeled as “Host Name”.
C. Text field labeled as “Finding Type”.
D. Text field labeled as “Description”.
E. Text field labeled as “Status”.
F. Text field labeled as “Posture”.
G. Text field labeled as “C”.
H. Text field labeled as “I”.
I. Text field labeled as “A”.
J. Text field labeled as “C”.
K. Text field labeled as “I”.
L. Text field labeled as “A”.
M. Text field labeled as “CAT Score:”.
N. Text field labeled as “CM”.
O. Text field labeled as “VS(n)”.
P. Text field labeled as “VS(q)”.
Q. Text field labeled as “Relevance of Threat”.
R. Text field labeled as “Likelihood”.
S. Text field labeled as “Impact”.
T. Text field labeled as “Imp Score”.

U. Text Box labeled as “Long Description”.
V. Text Box labeled as “Impact Rationale”.
W. Text Box labeled as “Mitigation”.
X. Text Box labeled as “Analyst Initials”.
Y. Button labeled as “Close”.

[SRS 14] The following Edit Sub-Task should contain the following components in Figure 14.1. This will allow a lead analyst to edit a currently existing task, by changing all the components provided.

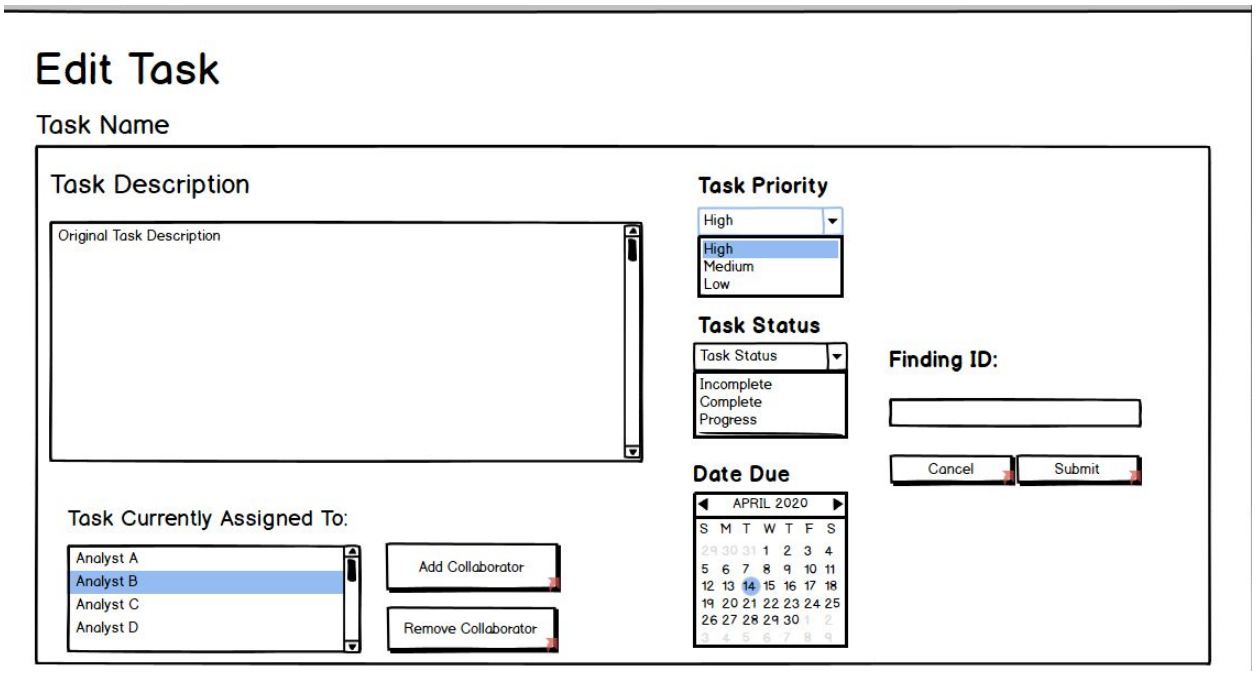


Figure 14.1:Edit Task.

A. Overlay title labeled as “Edit Tasks”.
B. Layout named “Task Name”.
C. Text Box Labeled as “Task Description”.
D. Drop down menu labeled as “Task Priority”.
E. Option inside drop down menu listed as “High”.
F. Option inside drop down menu listed as “Medium”.
G. Option inside drop down menu listed as “Low”.
H. Text Input Box labeled as “Finding ID”.
I. Drop down menu labeled as “Task Status”.
J. Option inside drop down menu listed as “Incomplete”.
K. Option inside drop down menu listed as “Complete”.
L. Option inside drop down menu listed as “In Progress”.
M. Selection List labeled as “Task Currently Assigned To”.
N. Button labeled as “Add Collaborator”.
O. Button labeled as “Remove Collaborator”.

P. Calendar Selector labeled as “Date Due”.
Q. Button Labeled as Cancel.
R. Button Labeled as Submit.

[SRS 15] The following Sub-Tasks should contain the following components in Figure 15.1. This shall provide information of a subtask that is one the system.

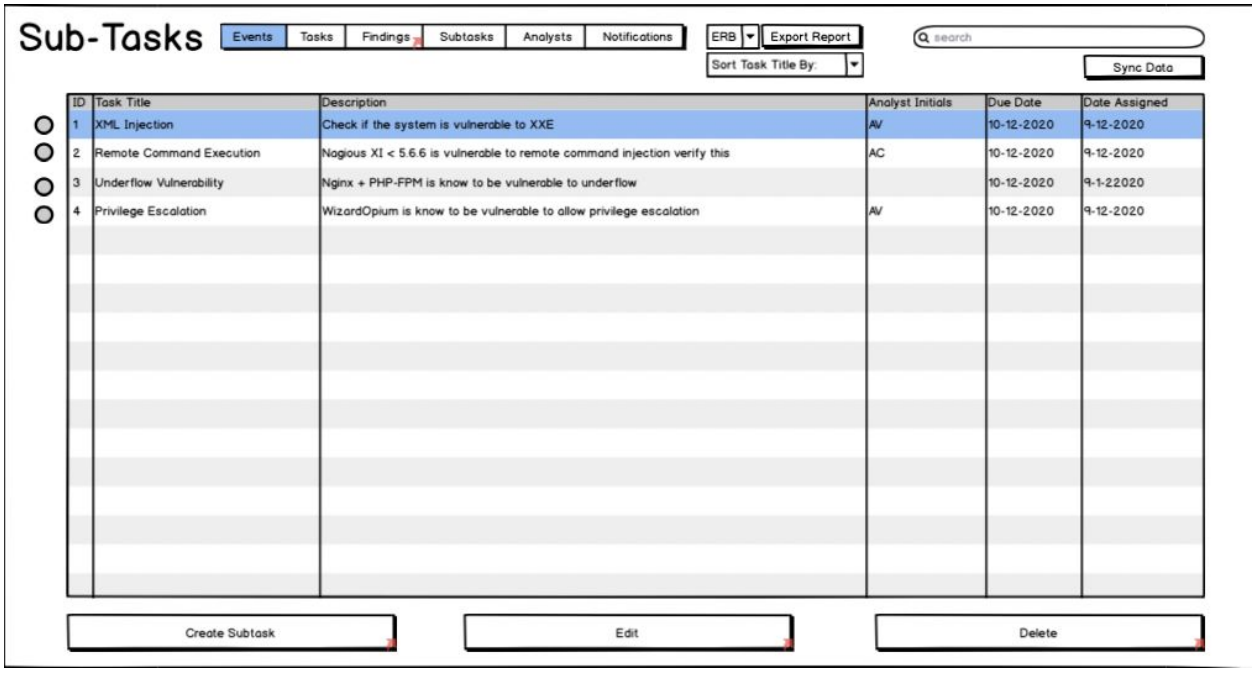


Figure 15.1: Sub-Tasks

A. Overlay title labeled as Sub-Tasks.
B. Navigation bar to navigate to other views through the use of buttons including:
    a. Events,
    b. Tasks,
    c. Findings
    d. Subtasks.
    e. Analyst.
    f. Notifications.
C. Dropbox labeled as “ERB, Risk Matrix, and Final Report”.
D. Button labeled as “Export Report”.
E. Dropbox labeled as “Sort Task Title By”.
F. Button labeled as “Sync Data”.
G. Button labeled as “Create Subtask”.
H. Button labeled as “Edit”.
I. Button labeled as “Delete”.
J. Search box labeled as “search”.
K. Check bubbles help in selecting/deselecting a task .
L. Text toolbar that contains Sub-Task information.

a. ID.
b. Task Title.
c. Description.
d. Analyst Initials.
e. Due Date.
f. Date Assigned.

[SRS 16] The following Remove Collaborators should contain the following components in Figure 16.1 The following shall show the analyst a window to remove a collaborator from the current project.

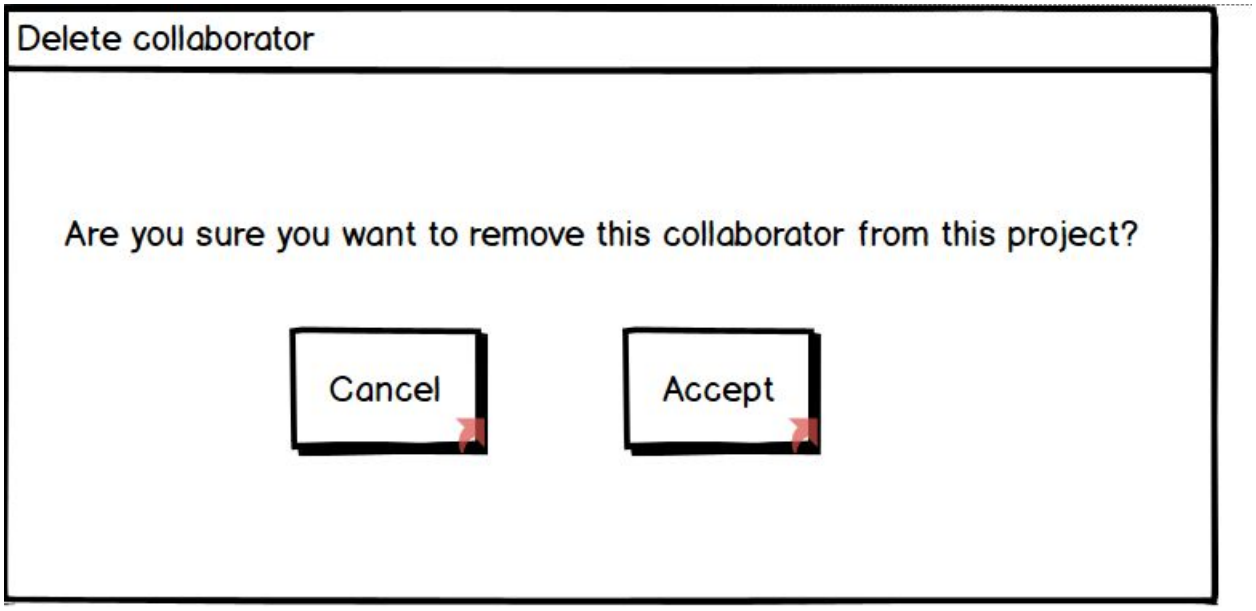


Figure 16.1: Delete collaborator.

A. Overlay title labeled as Remove Collaborator.
B. Text labeled as “Are you sure you want to remove this collaborator from this project?”.
C. Button labeled as “Cancel”.
D. Button labeled as “Accept”.

[**SRS 17**] The following Delete subtask should contain the following components in Figure 17.1. This shall provide a way to delete a subtask.

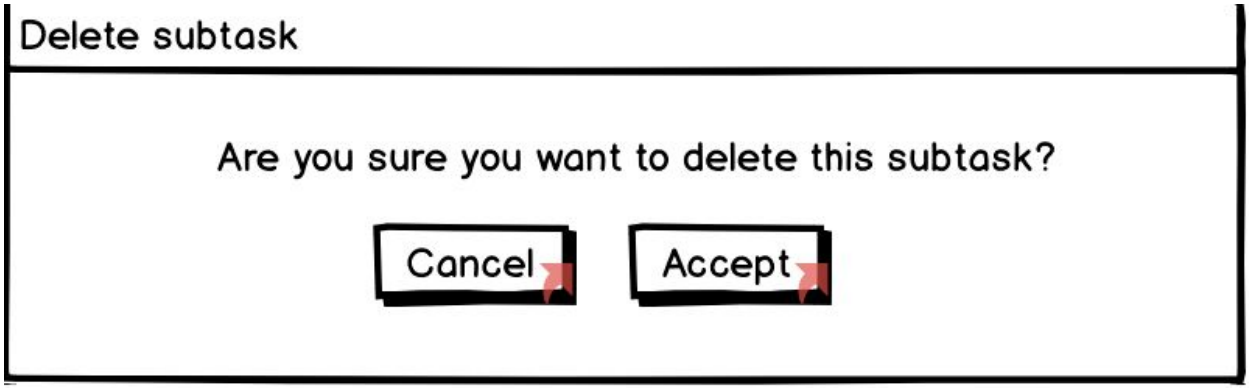


Figure 17.1: Delete Subtask.

A. Overlay title labeled as Delete subtask.
B. Text labeled as “Are you sure you want to delete this subtask?”.
C. Button labeled as “Cancel”.
D. Button labeled as “Accept”.

[SRS 18] The following Create Sub-Task should contain the following components in Figure 18.1 The system will allow the analyst to create a sub-task.

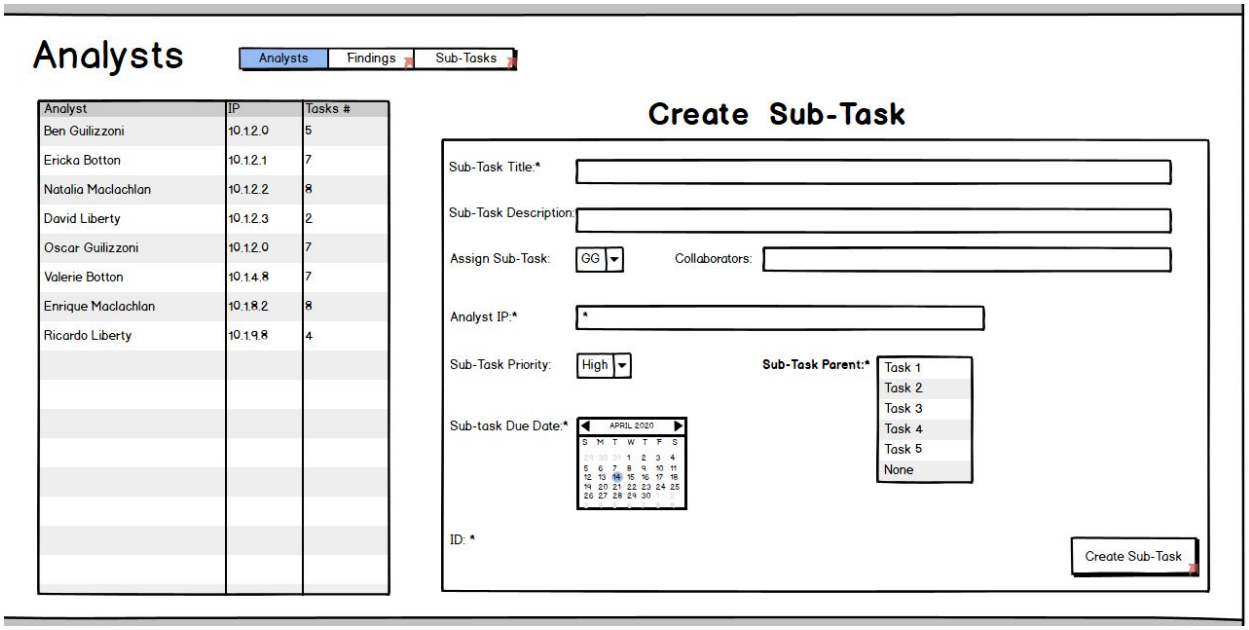


Figure 18.1 Create Sub-Task

A. Overlay title labeled as “Analyst”.
B. Navigation bar to navigate to other views through the use of buttons including:
    a. Analyst.
    b. FIndings.
    c. Sub-Tasks.

C. Table that contains the the analysts to assign a Sub-Task to:
    a. Analyst.
    b. IP.
    c. Task #.
D. Layout labeled as “Create Sub-Task”.
E. Text box labeled as “Sub-Task Title”.
F. Text box labeled as “Sub-Task Description”.
G. Drop down menu labeled as “Assign Sub-Task”.
H. Text box labeled as “Collaborators”.
I. Text box labeled as “Analyst IP:”.
J. Drop down menu labeled as “Sub-Task Priority”.
K. Calendar Selector labeled as “Sub-Task Due Date”.
L. List of tasks labeled as “Sub-Task Parent”.
M. Text field labeled as “ID”.
N. Button labeled as “Create Sub-Task”.

[SRS 19] The following Edit Sub-Task should contain the following components in Figure 19.1. The system will allow the analysts to edit currently existing tasks by modifying the following components.

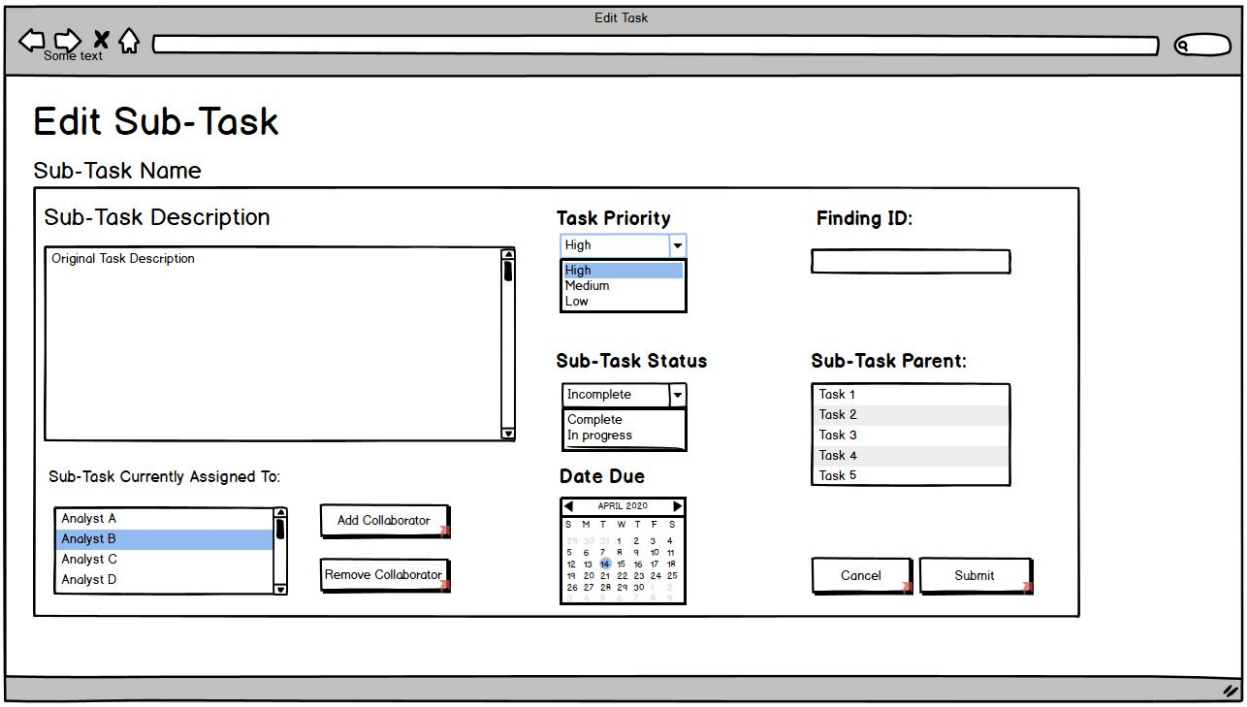


Figure 19.1:Edit Sub-Task

A. Overlay title labeled as “Edit Sub-Tasks”.
B. Layout named “Sub-Task Name”.
C. Text Box Labeled as “Sub-Task Description”.
D. Drop down menu labeled as “Task Priority”.

E. Text Input Box labeled as “Finding ID”.
F. Drop down menu labeled as “Sub-Task Status”.
G. List labeled as “Sub-Task Parent”.
H. Selection List labeled as “Sub-Task Currently Assigned To”.
I. Button labeled as “Add Collaborator”.
J. Button labeled as “Remove Collaborator”.
K. Calendar Selector labeled as “Date Due”.
L. Button Labeled as Cancel.
M. Button Labeled as Submit.

[SRS 20] The following Search Result should contain the following components in Figure 20.1. The system shall display related findings,tasks,and sub-task.

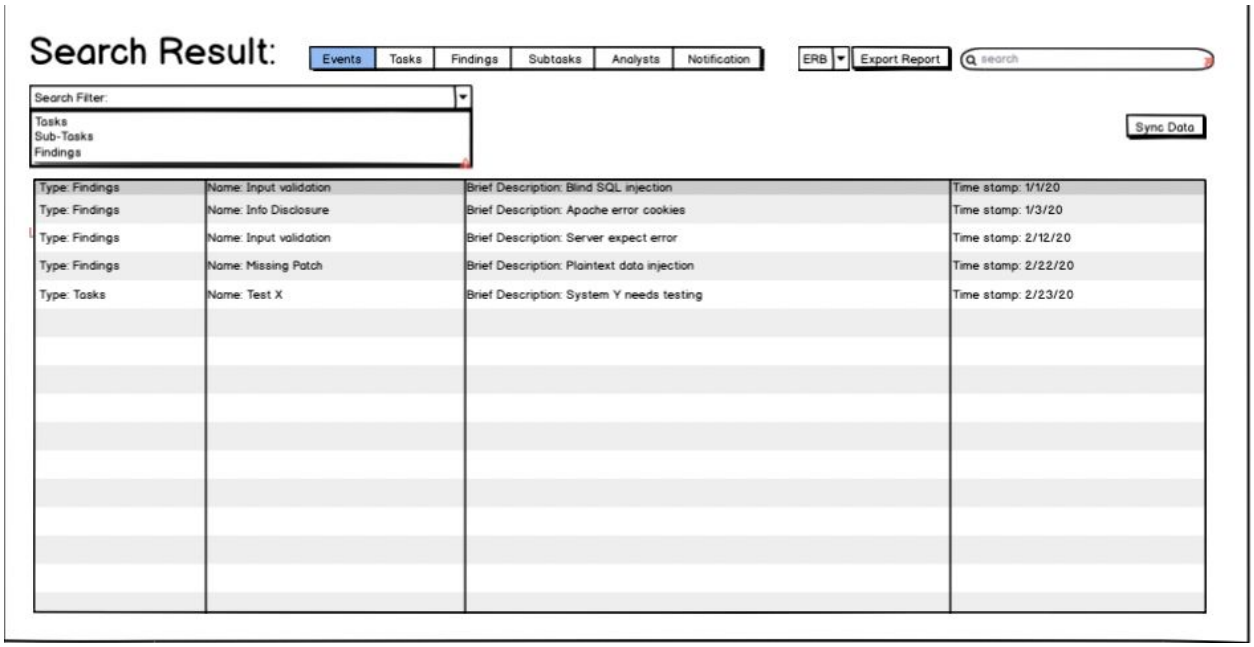


Figure 20.1:Search Result page

A. Overlay title labeled as Search Result.
B. Button Toolbar labeled as “Events, Tasks, Findings, Subtasks, Analyst, and Notifications”.
C. Dropbox labeled as “ERB, Risk Matrix, and Final Report”.
D. Button labeled as “Export Report”.
E. Button labeled as “Sync Data”.
F. Search box labeled as “search”.
G. Dropbox labeled as “Search Filter”.
    a. Button labeled as “Tasks”.
    b. Button labeled as “Sub-Tasks”.
    c. Button labeled as “Findings”.

H. Text toolbar labeled as “Type, Name, Brief Description, and Time stamp”.

[SRS 21] The following Results should contain the following components in Figure 21.1. This shall display the information based off the search result.

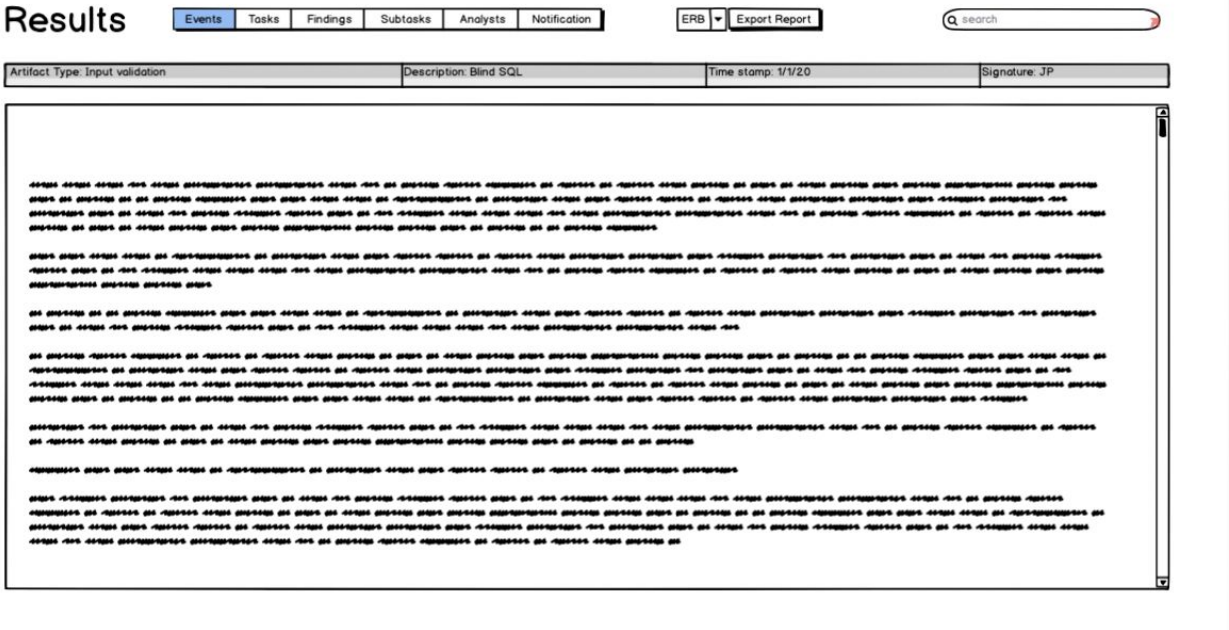


Figure 21.1: Searched Result

A. Overlay title labeled as Result.
B. Button Toolbar labeled as “Events, Tasks, Findings, Subtasks, Analyst, and Notifications”.
C. Dropbox labeled as “ERB, Risk Matrix, and Final Report”.
D. Button labeled as “Export Report”.
E. Search box labeled as “search”.
F. Text toolbar labeled as “Artifact Type: , Description: ,Time stamp: , Signature: ”.
G. TextField displays the information of the result searched.
    a. Scrollbar helps with navigating up/down in the textbox for further information.

[SRS 22] The following Notification should contain the following components in Figure 22.1

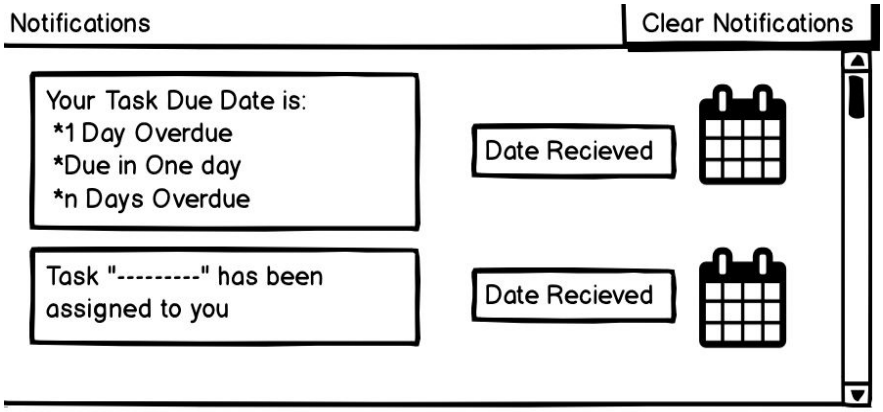


Figure 22.1:Notifications Page.

A. Overlay title labeled as Notifications.
B. Button labeled as “Clear Notifications”.
C. TextBox labeled as “Your Task Due Date is:”.
D. TextBox labeled as “Task “---” has been assigned to you”.
E. Button labeled as “Date Received”.
F. Button labeled as “Date Received”.
G. CalanderPick provides a way to select dates in an ordered manner.
H. CalanderPick provides a way to select dates in an ordered manner.
I. Scrollbar provides a way to navigate up/down in SRS[26].

[SRS 23] The system shall have a setting page with the following components as shown in Figure 23.1.

Both Lead Analysts and Analyst shall be able to adjust the system text size and identify if need need a color blind friendly view.

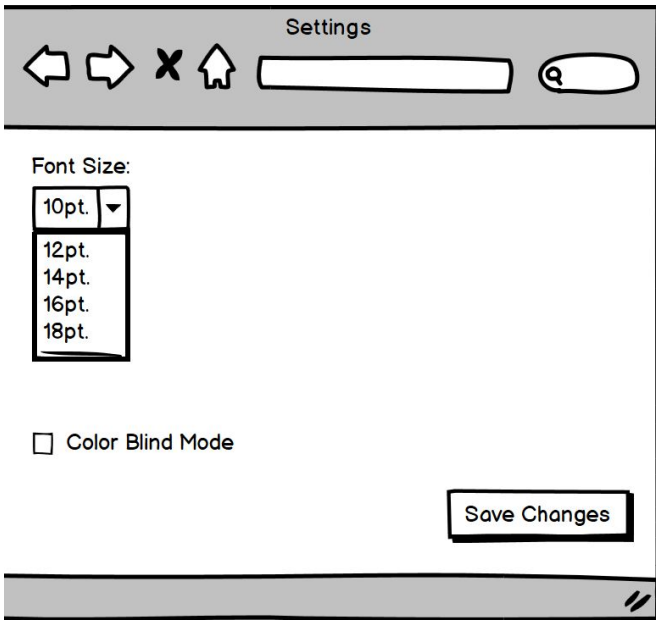


Figure 23.1:Setting Homepage.

A. A web browser to indicate the system is a web based application
B. Overlay title named “Settings”.
C. A drop down button labeled “Font Size”.
D. A checkbox labeled “Color Blind Mode”.
E. A button labeled “Saved Changes”.

[SRS 24] The system shall display an error message with the following components as shown in figure 24.1. The system shall display an error message when required text fields are missing when creating a Task, Subtask, and Finding.

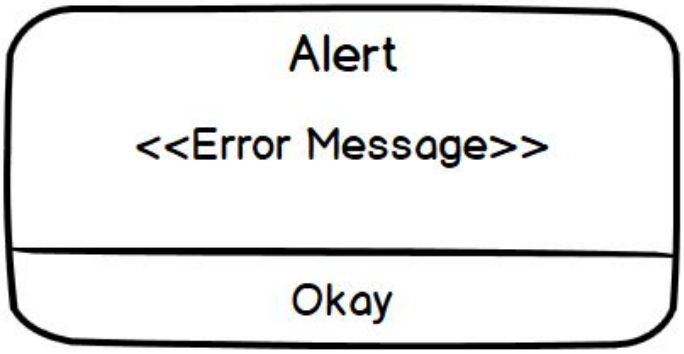


Figure 24.1:Alert Notification

A. A pop up notification box labeled “Alert”.
B. A text field displaying the unique error.
C. A button labeled “Okay” to acknowledge the user has seen the notification.

[SRS 25] The system shall display a pop up Delete Task page which contains the following components as shown in Figure 25.1
The system shall make sure that a Lead Analyst wants to remove the given task.

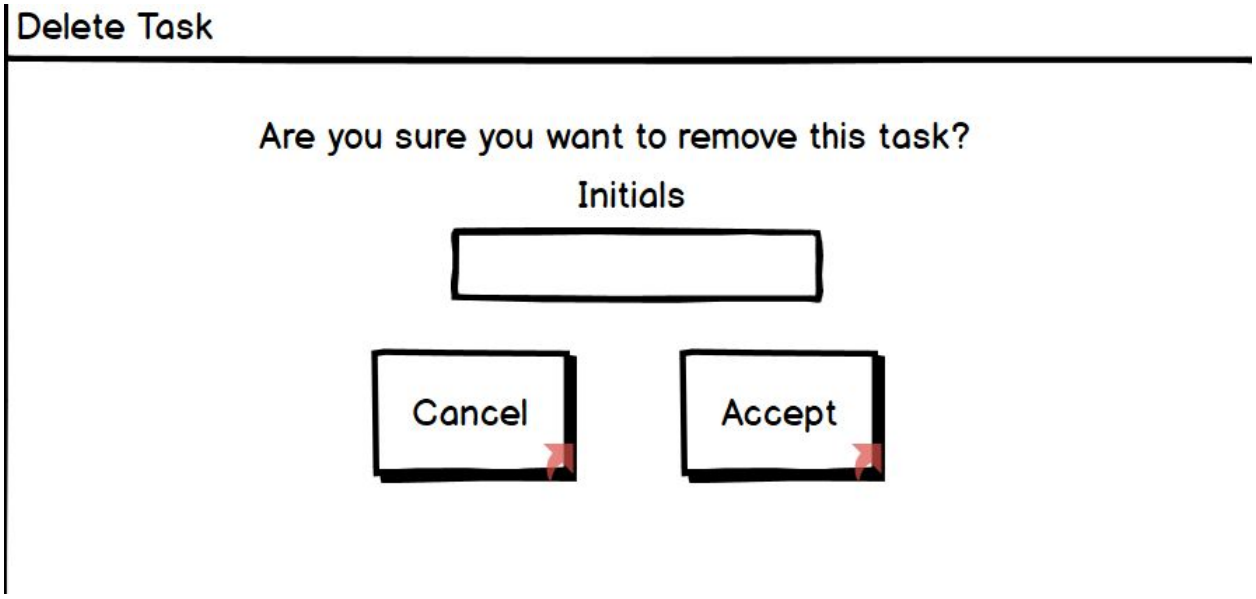


Figure 25.1:Delete Task Pop Up.

A. A text title named “Delete Task”.
B. A text field named “Are you sure you want to remove this task?”.
C. A text box named “Initials”.
D. A button named “Cancelled”.
E. A button named “Accept”.

[SRS 26] The system shall display a Task Homepage which contains the following components as shown in Figure 26.1
The system shall display all current tasks in the event with the following information:

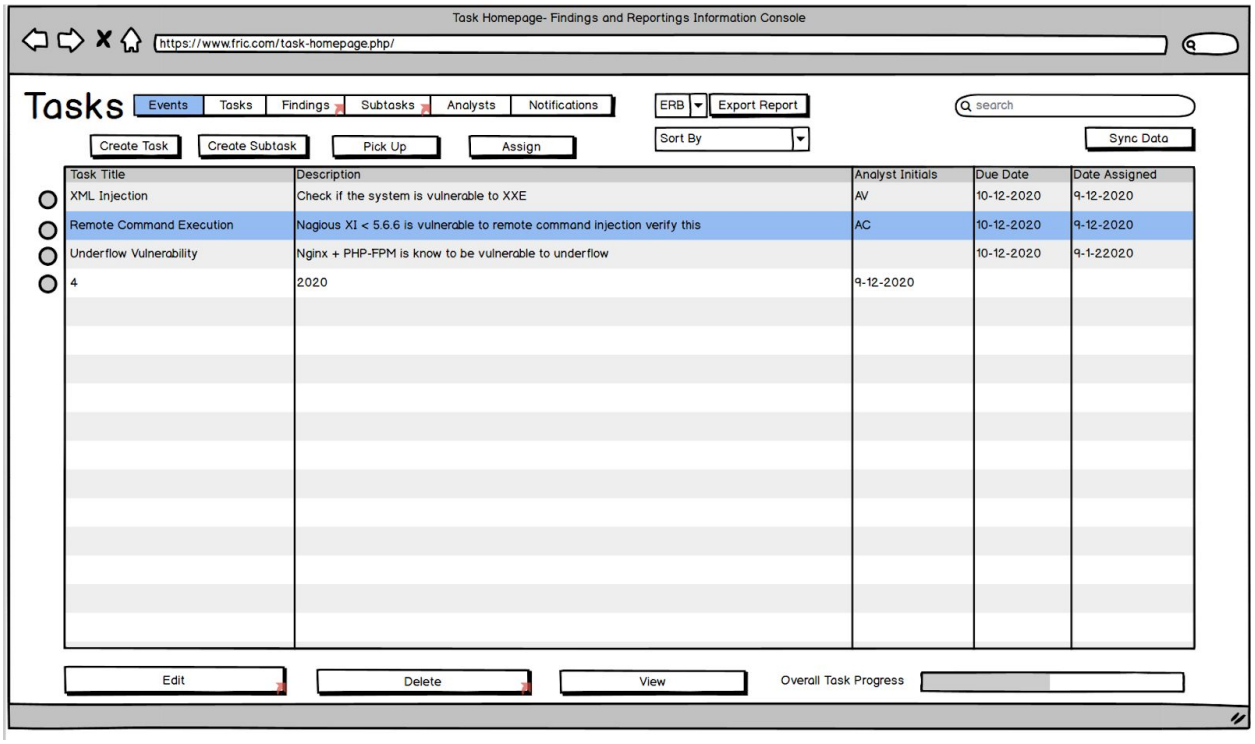


Figure 26.1:Task Homepage.

A. A text overlay title named “Tasks”
B. Navigation bar to navigate to other views through the use of buttons including:
    a. A button named “Events”.
    b. A button named “Tasks”.
    c. A button named “Findings”.
    d. A button named “Subtasks”.
    e. A button named “Analysts”.
    f. A button named “Notifications”.
C. A button named “Create Task”.
D. A button named “Create Subtask”.
E. A button named “Pick Up”.
F. A button named “Assign”.
G. A dropdown button for the 3 different reports.
H. A button named “Export Report”.
I. A dropdown button named “Sort By”.
J. A button named “Sync Data”.
K. A search bar named “search”.
L. A table with the following information:
    a. Task Title.
    b. Task Description.
    c. Analyst Initials.
    d. Due Date.

e. Date Assigned.

M. A check bubble for each task present.
N. A button named “Edit”.
O. A button named “Delete”.
P. A button named “View”.
Q. A progress bar named “Overall Task Progress”

3.1.2. Hardware Interfaces

Currently no hardware interface requirements are listed by the clients.

3.1.3. Software Interfaces

[SRS 27] The system shall run on a Kali Linux operating system.

3.1.4. Communications Interfaces

[SRS 28] The system shall be encrypted on a closed LAN network.

3.2. Behavioral Requirements

This section introduces the behavior of each entity in F.R.I.C. The entities are analyst and lead analyst. The two entities differ in their permissions to edit and delete artifacts from F.R.I.C. Each requirement is grouped by the entity and describes what each entity is allowed to do.

3.2.1. Same Class of User

There are seven types of access in the system:

I. Write.
II. Read.
III. Append.
IV. Associate.
V. Sync.
VI. Export.
VII. Promote / Demote

[SRS 29] The system shall have three levels of access privileges:

I. Lead Analyst.
II. Analyst.
III. Collaborator.

Lead Analyst Permission Requirements

[SRS 30] The system shall give write access to a lead analyst to create:

I. Systems
II. Tasks
III. Findings
IV. Event

[SRS 31] The system shall give write access to a lead analyst to edit:

I. Systems
II. Tasks
III. Findings
IV. Event

[SRS 32] The system shall give write access to a lead analyst to delete:

I. Systems
II. Tasks
III. Findings

IV. Events

[SRS 33] The system shall give associate access to a lead analyst to assign:

I. Tasks
II. Subtasks

[SRS 34] The system shall give associate access lead analysts to relate ___ to a system(s).

I. Tasks
II. Subtasks

[SRS 35] The system shall give associate access to a lead analyst to allow analysts to pick up unassigned tasks.

I. The lead analyst has the option to assign a task or let it be picked up.

[SRS 36] The system shall give sync access to a lead analyst to sync to:

I. Lead analyst
II. Specific analyst
III. All analysts

[SRS 37] The system shall give read access to a lead analyst to view the progress of ___.

I. Task
II. Subtask
III. Event
IV. System

[SRS 38] The system shall give export access to the lead analyst for exporting a formatted technical.

[SRS 39] The system shall give sync access to the lead analyst for being able to directly sync with an analyst.

[SRS 40] The system shall give write access to allow a Lead Analyst to mark themselves as a lead.

[SRS 41] The system shall give write access to the lead analyst for deleting a whole event.

[SRS 42] The system shall give append access to the original lead analyst to demote other leads to regular analysts.

[SRS 43] The system shall give append access to a lead analyst to promote an analyst to be a lead analyst for a particular event.

[SRS 44] The system shall give an analyst append access to a related finding to a __

I. Task
II. Subtask

Analyst Permission Requirements

[SRS 45] The system shall give an analyst append access to a related finding to a ___

I. Task
II. Subtask

[SRS 46] The system shall allow an analyst to have read access to the status of ___.

I. Task
II. Subtask

[SRS 47] The system shall give write access to an analyst to ___ a finding.

I. Create
II. Edit
III. Delete

[SRS 48] The system shall give associate access to an analyst to associate a finding to any other finding.

[SRS 49] The system shall give an analyst read access to view a finding that they have not created themselves.

[SRS 50] The system shall give read access to an analyst to search for a finding.

[SRS 51] The system shall give an analyst read access to sort findings by finding attributes.

[SRS 52] The system shall give read access to an analyst to sync their data with any other analyst

[SRS 53] The system shall give write access to allow an analyst to create a sub-task.

[SRS 54] The system shall give associate access to an analyst to relate a subtask to a task.

[SRS 55] The system shall give write access to an analyst to edit subtasks that they have created.

[SRS 56] The system shall give write access to allow an analyst to delete sub-tasks that they created.

[SRS 57] The system shall give write access to analysts to delete their own findings.

[SRS 58] The system shall allow write access to an analyst to upload files to support their finding.These files included but are not limited to:

I. .pcap files
II. .jpeg files
III. .mp4 files
IV. .ova files

[SRS 59] The system shall give an analyst export access when generating the following reports:

I. Risk Matrix Report
II. ERB Report
III. Final Report

[SRS 60] The system shall allow analysts write access to tasks assigned to them only to change the status of those tasks. An analyst cannot change the task status of a task not assigned to them.

[SRS 61] The system shall allow an analyst write access to edit a task status when the following happens:

I. A lead analyst assigns a task.
II. An analyst picks up a task.

[SRS 62] The system shall give sync access to an analyst to sync to :

I. Lead analyst
II. Specific Analyst
III. All analysts

3.2.1.1. Access Table

Append - A
Associate- S
Write W
Export - X
Read - R
Sync- N
Promote/Demote - P

	Finding	Task	Subtask	Report	Promote/Demote
Lead	A,S,W,X,R,N	A,S,W,P,R	A,W,P,R	X,N	P
Analyst	S,W,R,N,A	R	S,W,R,P,A	X,N	-
Collaborator	A,R	A,R	A,R	-	-

The section below describes the details of the relationships between inputs that exist in F.R.I.C. The section is broken down into the Real World Objects that exist in F.R.I.C. There are 9 objects represented by tables in which attributes, data types, constraints, and a general description are listed. These are all of the relationships that exist in the current version of F.R.I.C.

3.2.2. Related Real-World Objects

[SRS 63] The system shall store the attributes as defined in Table 1 for a task configuration.

Table 1: Task configuration

Attribute	Data Type	Values and Constraints	Description
End date	Date	Required; Editable; MM/DD/YYYY and D-D/MM/YYYY	Deadline of the task.
Start date	Date	Required;Editable; MM/DD/YYYY and D-D/MM/YYYY	Starting date of the task.
description	String	Required;Editable;	Description of the task
Title	String	Required;Editable;	Name of the task.
Priority	String	Required;Editable; (High, Medium, Low)	Urgency to complete the given task
Assigned analyst	Object: Analyst	Editable;	Designated analyst for the given task.
Progress	String	Require; (not-doable, not started, assigned, transferred,	Percentage of the completion of the task.

		in-progress, complete);	
Collaborators	Object: Analyst	Required;Editable;	External analyst to the task.
Comment	String	Editable;	Side notes/observations made by the analyst.
Child task	Object: Subtask	Editable;	Subtask linked to task.

[SRS 64] The system shall store the attributes as defined in Table 2 for a subtask configuration.

[SRS 65] The system shall allow the user to associate subtasks with tasks.

[SRS 66] The system shall allow the user to have multiple findings associated with tasks.

Table 2: Sub-Task configuration

Attribute	**Data Type**	**Values and Constraints**	**Description**
End date	Date MM/DD/YYYY and D-D/MM/YYYY	Required;Editable;	End date of the subtask.
Start date	Date MM/DD/YYYY and D-D/MM/YYYY	Required;Editable;	Start date of the subtask.
Description	String	Required;Editable;	Description of the subtask
Title	String	Required;Editable;	Title of the subtask
Priority	String	Required;Editable;	Urgency to complete the given subtask

Progress	String	Require; (not-doable, not started, assigned, transferred, in-progress, complete)	Urgency to complete the given subtask
Comment	String	Editable;	Side notes/observations made by the analyst.
Collaborator	Object: Analyst	Editable;	External analyst to the subtask
Parent	Object: Task	Required; Editable;	Subtask that stems from a given task

[SRS 67] The system shall store the attributes as defined in Table 3 for a system configuration.
[SRS 68] The system (F.R.I.C.) shall allow a system to have one or more tasks.

Table 3: System configuration

Attribute	**Data Type**	**Values and Constraints**	**Description**
Host Name	String	Required, Editable	The name of the host that is a part of the system under test.
IP Port	String	Required, Editable	The connection endpoint of the system.
System Description	String	Required, Editable	A detailed description of the system being tested
Confidentiality	String	(Very High,High, Moderate, Low, very Low)	Level of unauthorized access achieved
Integrity	String	(Very High,High, Moderate, Low, very	Level of unauthorized data alteration.

		Low)	
Availability	String	(Very High,High, Moderate, Low, very Low)	Level of unauthorized user access.
Location	String	Require,Editable	Location of the system(s)being tested
Building Room Accessed	Integer	Require,Editable	Precise building and room number that the system is being tested on
Assessment Dates	Date	Required, Editable	The range of dates the system was evaluated.

[SRS 69] The system shall store the attributes as defined in Table 4 for an event configuration.
[SRS 70] The system (F.R.I.C.) shall have an event that has one or more systems.

Table 4: Event configuration

Attribute	**Data Type**	**Values and Constraints**	**Description**
Event Name	String	Required; Editable	Name of the current event.
Event Description	String	Required; Editable	Description of the current event.
Assessment Dates	Date MM/DD/YYYY and D-D/MM/YYYY	Required; Editable; D-D/MM/YY	Range of dates that the event will take place.

Tested Systems	String	Required, Editable	The systems that are to be tested in an event/have already been tested
Event Type	String	Required, Editable (CPVA, AA, CVII)	The type of assessment that will be taking place
Version	Double	Required, Editable	Keeping track of the number of revisions of the Event the team went through
Security Classification Guide Title	String	Required, Editable	The classification title of the system being tested
Declassification Date	Date	Required; Editable; MM/DD/YY	Day that information can be released to those who did not previously have access
Organization Name	String	Required, Editable	Name of the organization that client is doing penetration testing for
Customer Name	String	Required, Editable	Name of the company requesting the assessment

[SRS 71] The system shall store the attributes as defined in Table 5 for a report configuration.

Table 5: Report configuration.

Attribute	Data Type	Values and Constraints	Description
System Under Attack	String	Required, Editable	The system(s) that were penetration tested
Level of Access Achieved	String	Required, Editable	Description of the Level of penetration achieved during analysis.
Process	String	Required, Editable	A description of the the process the analyst went through to discover the finding(s)

[SRS 72] The system shall store the attributes as defined in Table 6 for a finding configuration.

Table 6:Finding configuration.

Attribute	Data Type	Values and Constraints	Description
ID	Integer	Required; Editable	The unique identifier for a finding.
Status	String	Required; Editable (Open, Closed)	The current standing of the finding.
Host Name	String	Required Editable	A label that identifies a device that is connected to a local network.

Type	String	Required; (CREDENTIALS COMPLEXITY ,MANUFACTURER DEFAULT CREDS ,LACK OF AUTHENTICATION, ,PLAIN TEXT PROTOCOLS, PLAIN TEXT WEB-LOGIN, ENCRYPTION, AUTHENTICATION BYPASS, PORT SECURITY, ACCESS CONTROL, LEAST PRIVILEGE, PRIVILEGE ESCALATION, MISSING PATCHES, PHYSICAL SECURITY)	Identifies the class of finding
Posture	String	Required, Editable (Insider, Outsider, Nearsider, Nearsider-Outsider,Insider-Nearsider)	Refers to the level of access the analyst uses to attack the system.
Relevance	String	Required, Editable (Confirmed, Expected, Anticipated, Predicted, Possible)	The status of the severity of a finding.
Analyst	Object: Analyst	Required, Editable	The initials of the analyst who documented the finding.

Description	String	Required, Editable	The steps taken to produce the vulnerability in 1 line of characters.
Long Description	String	Required, Editable	The steps taken to produce the vulnerability.
Confidentiality	String	(Very High,High, Moderate, Low, very Low)	Level of unauthorized access achieved
Integrity	String	(Very High,High, Moderate, Low, very Low)	Level of unauthorized data alteration.
CAT	Integer	(CAT I, CAT II, CAT III)	A measure of vulnerability severity.
CAT Score	Integer	Required; Derived {10,7,4}	A calculation to measure the vulnerability severity
Relevance of Threat	String	(Confirmed, Expected, Antificapted Predicted, Possible), Derived	The predicted level of danger.
Availability	String	(Very High,High, Moderate, Low, very Low)	Level of unauthorized user access.
Likelihood	String	(Very High,High, Moderate, Low, very Low) Derived	The probability of the vulnerability having an effect.
Risk	Integer	Required, Editable, Derived (Very High,High, Moderate, Low, very Low) Derived	Probability of exposure

Impact	Integer	Required, Editable, Derived (Very High,High, Moderate, Low, very Low) Derived	Severity of damage the vulnerability will allow.
Impact Rationale	String	Required, Editable	A description of why this level of impact was chosen.
Mitigation	String	Required, Editable	Description on reducing or assessing the level of threat.
MitigationOneLiner	String	Required, Editable	One line description on reducing the level of threat.
Countermeasure	Integer	Required, Editable, Derived	Effectiveness of a countermeasure against a discovered vulnerability
Artifacts(Attachments)	String	Editable	Attached files that support the given finding
VS Score	String	Required; Derived; (Very,High,High, Moderate, Low, Very Low)	Vulnerability Severity Score

[SRS 73] The system shall store the attributes as defined in Table 7 for analyst.
[SRS 74] The system shall allow an analyst to have a task.
[SRS 75] The system shall allow for an analyst to create findings.

Table 7: analyst configuration.

Attribute	Data Type	Value and Constraints	Description
Initials	String	Required, Editable	For analyst signature for a task
IP Address	String	Required, Editable	Will be used to create a unique login
Current Tasks	Task	Required, Editable	Display the selected task from the analyst.
Notifications	Object : Notification	Required, Editable	Shall sends alert due date of an upcoming task

[SRS 76] The system shall store the attributes as defined in Table 8 for Lead analyst configuration.
[SRS 77] The system shall allow the lead analyst to create an event.

Table 8: Lead Analyst configuration.

Attribute	Data Type	Value and Constraints	Description
Initials	String	Required, Editable	For analyst signature for a task.
Ip Address	String	Required, Editable	Will be used to create a unique login.
Current Tasks	Task	Required, Editable	Display the selected task from the analyst.
Notifications	Object: Notification	Required, Editable	Shall send an alert due date of an upcoming task.
IsLead	Boolean	Required, Editable	Will check if the analyst is a lead for a certain event.
Task List	Task	Required, Editable	All of the existing tasks

[SRS 78] The system shall store the attributes as defined in Table 9 for notifications.

[SRS 79] The system shall send a notification to a lead analyst if:

1. An analyst has a past due task
2. An analyst updates the status of a task

[SRS 80] The system shall send a notification to a lead analyst if:

1. The have a task that has not been completed a day before it is due
2. They have a task that is overdue

Table 9: Notifications

Attribute	**Data Type**	**Value and Constraints**	**Description**
Lead Analyst	Object: Lead Analyst	Required	The initials of the lead analyst to identify the recipient of the notification.
Analyst	Object: Analyst	Required	The initials of the analyst to identify the recipient of the notification.
Task	Object: Task	Required	The information of the task that has triggered a notification.
Subtask	Object: Task	Required	The information of a subtask that has triggered a notification.

3.2.3. Stimulus

This section describes the user interface and system stimulus requirements. Stimulus requirement describes how the specific part of the system is supposed to react to other interactions. A state diagram will be provided to help visualize the systems reactions. A state diagram shows the conditions and events that take place to get the system from state A to state B. This includes the conditions that need to be met, the events that need to take place and the flow from state to state.

3.2.3.1. User Interface Stimulus

[SRS 81] When the user presses the login button, the system shall verify the user then display the event homepage if valid, and repeat if the user login is invalid.
[SRS 82] When the user presses the edit button on the event homepage, the system shall display the edit event page.
[SRS 83] When the user presses the add button on the event homepage, the system shall display the add event page.
[SRS 84] When the user presses the delete button on the event homepage, the system shall display a verification page.
[SRS 85] When the user presses the task button, the system shall display the task homepage.
[SRS 86] When the user presses the subtask button, the system shall display the subtask homepage.
[SRS 87] When the user presses the search button, the system shall display the search result homepage.
[SRS 88] When the user presses the edit button on the task homepage, the system shall display the edit task page.
[SRS 89] When the user presses the add button on the task homepage, the system shall display the add task page.
[SRS 90] When the user presses the notification button on the task homepage, the system shall display the upcoming due dates.
[SRS 91] When the user presses the button “Create Finding” the system shall display a “Create Findings Page”
[SRS 92] When the user attempts to create an invalid finding inside the “Create Findings Page” an error message will display “Your missing information. Please ensure all fields marked with * are filled.”
[SRS 93] When the user attempts to edit a finding inside the “Edit Findings Page” and leaves a field empty an error message will display “Your missing information. Please ensure all fields marked with * are filled.”.
[SRS 94] When the user presses the edit button ‘Edit’ the system shall display a “Edit Findings Page”
[SRS 95] When the user pressed the archive button, the system shall display a “Confirmation Page” asking the user to confirm their action.

3.2.3.2. System Stimulus

This section presents requirements that describe behavior the system exhibits when provided with a system stimulus. System stimulus requirements should be decoupled from the user interface so that when the interface changes, it will have minimal impacts on the system stimulus requirements. Requirements in this section are typically derived from a system-based or life cycle object-based state transition diagram.
[Reference the diagrams in the appendices.]

3.2.3.2.1. Finding

[SRS 96] When the user is in the create finding operation, the system shall do the following:

a. Gather user input for the finding

[SRS 97] When the user is in the create finding operation, after pressing “submit button”, the system shall do the following:

b. Gather any supporting file the user wishes to attach
c. Calculate all derived attributes this includes:
    i. Impact Score
    ii. CAT Score
    iii. Risk
    iv. Likelihood
    v. Vulnerability Severity

[SRS 98] When the user is in the create finding operation,and presses “submit” the system shall do the following.

a. Verify that all finding attributes have been filled out

[SRS 99] When the user is in the create finding operation, after the attributes have been verified(as stated in previous requirement), the system shall do the following:

a. Update the records to include the new finding

[SRS 100] When the user is in the edit finding operation,the system shall do the following:

a. The system will check if they have permission to edit the finding. A user has permission if they meet the following requirements:
    i. They are the owner of the finding
    ii. They are a lead analyst
b. If the user has permission the system will allow the user to update all fields of the finding

[SRS 101] When the user is in the edit finding operation, after entering the new finding information, the system shall do the following:

a. Verify that all finding attributes have been filled out
b. If the attributes have been verified the system will update the record of the finding

[SRS 101] When the user is in the archive finding operation the system shall do the following:

a. Load the finding the user wants to edit
b. Move the finding into archives
c. Delete the finding from the database

[SRS 102] When Archive Findings operation is complete, the system shall do the following:

a. Update the removal of the finding in the database

b. Log which analyst archived the finding

[SRS 103] When the user is in the export finding operation, after the user presses export,the system will do the following:

a. Gather all selected findings and tasks
b. Export them as specified by the user. The data has three different export files:
    i. ERB
    ii. Risk Matrix
    iii. Technical Report

3.2.3.2.2. Notification

[SRS 104] When a task due date is within the next day, the system shall do the following:
a. Notify the Lead analyst of the upcoming due date.
b. Notify the Analyst whom the task was assigned to of the upcoming due date

[SRS 105] When a task is overdue the system shall do the following:
a. Notify the Lead Analyst that the task if overdue
b. Notify the analyst whom the task was assigned to that the task is overdue

[SRS 106] When a subtask due date is within the next day, the system shall do the following:
a. Notify the Lead analyst of the upcoming due date.
b. Notify the Analyst whom the subtask was assigned to of the upcoming due date

[SRS 107] When a task is overdue the system shall do the following:
a. Notify the Lead Analyst that the subtask if overdue
b. Notify the analyst whom the subtask was assigned to that the subtask is overdue

3.2.3.2.3. Data Synchronization

[SRS 108] When the sync operation is complete, the following operations pertaining tasks information must be completed:
a. Task data must be pushed to the database
b. Task data must be pulled from the database

[SRS 109] When the sync operation is complete, the following operations pertaining subtasks information the system shall:
a. Sync subtask data to be pushed to the database
b. Sync subtask data to be pulled from the database

[SRS 110] When the sync operation is complete, the following operations pertaining findings information the system shall:
a. Sync finding data to be pushed to the database
b. Sync finding data to be pulled from the database

3.3. Non-Behavioral Requirements

The sections below describe the requirements that outline the quantifiable elements of F.R.I.C. The requirements will be split into six sections each describing the non-behavioral requirements in that area. The areas are: performance, qualitative, availability, security, maintainability, and portability.

3.3.1 Modifiability

QAS Scenario Modifiability Scenario: The analyst wants to change the font size in F.R.I.C.

Source: Analyst
Stimulus: Modify Font size
Artifact: “Settings” Page
Environment: Runtime
Response: Apply font changes
Response Measure: Font size

3.3.2 Usability

QAS Scenario Usability Scenario: An non-lead analyst or non-analyst tries to login to F.R.I.C.

Source: Non-lead analyst or non-analyst
Stimulus: Non-lead analyst or non-analyst attempts to login.
Environment: Runtime
Artifact: Login Page
Response: The system will not allow the non-lead analyst or non-analyst access to F.R.I.C.
Response Measure: The user has not gained any knowledge of the system.

QAS Scenario Usability Scenario: The analyst wants to edit an existing finding.

Source: Analyst
Stimulus: Analyst clicks Edit Finding.
Artifact: ‘Edit Findings’ page
Environment: Runtime
Response: The system shall display the Edit Findings page.
Response Measure: The time taken to display the Edit Findings page.

QAS Scenario Usability Scenario: An analyst wants to filter their search in the F.R.I.C. system.

Source: Analyst
Stimulus: The analyst enters a keyword to the ‘search filter’

Artifact: ‘Search Result’ Page
Environment: Runtime
Response: All artifacts in F.R.I.C. that contain the keyword are displayed.
Response Measure: Number of times the ‘search filter’ is pressed in comparison to number of clicks needed to manually find the artifact the analyst was looking for.

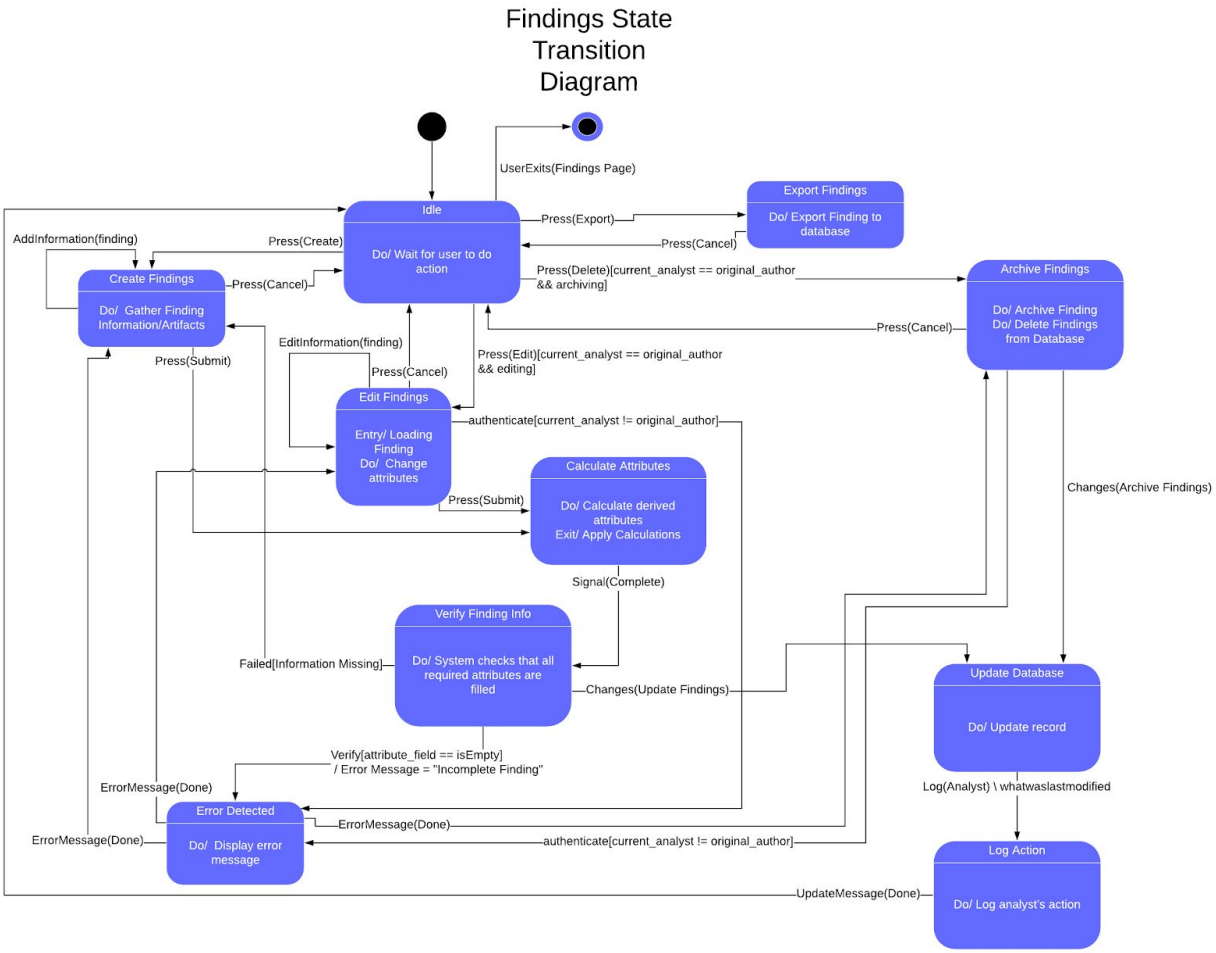
QAS Scenario Usability Scenario: An analyst wants to sync all data with other analysts.
Source: Analyst
Stimulus: The analyst presses the sync button.
Artifact: Home Page
Environment: Runtime
Response:All artifacts including findings, tasks, subtasks, and reports are updated with all analysts findings.
Response Measure: The time it took to sync all the data.

QAS Scenario Usability Scenario: An analyst wants to create a finding.
Source: Analyst
Stimulus: The analyst presses “Create Finding”
Artifact: Create finding page
Environment: Runtime
Response: The system will create a finding.
Response Measure: One new finding is created

4. Appendix

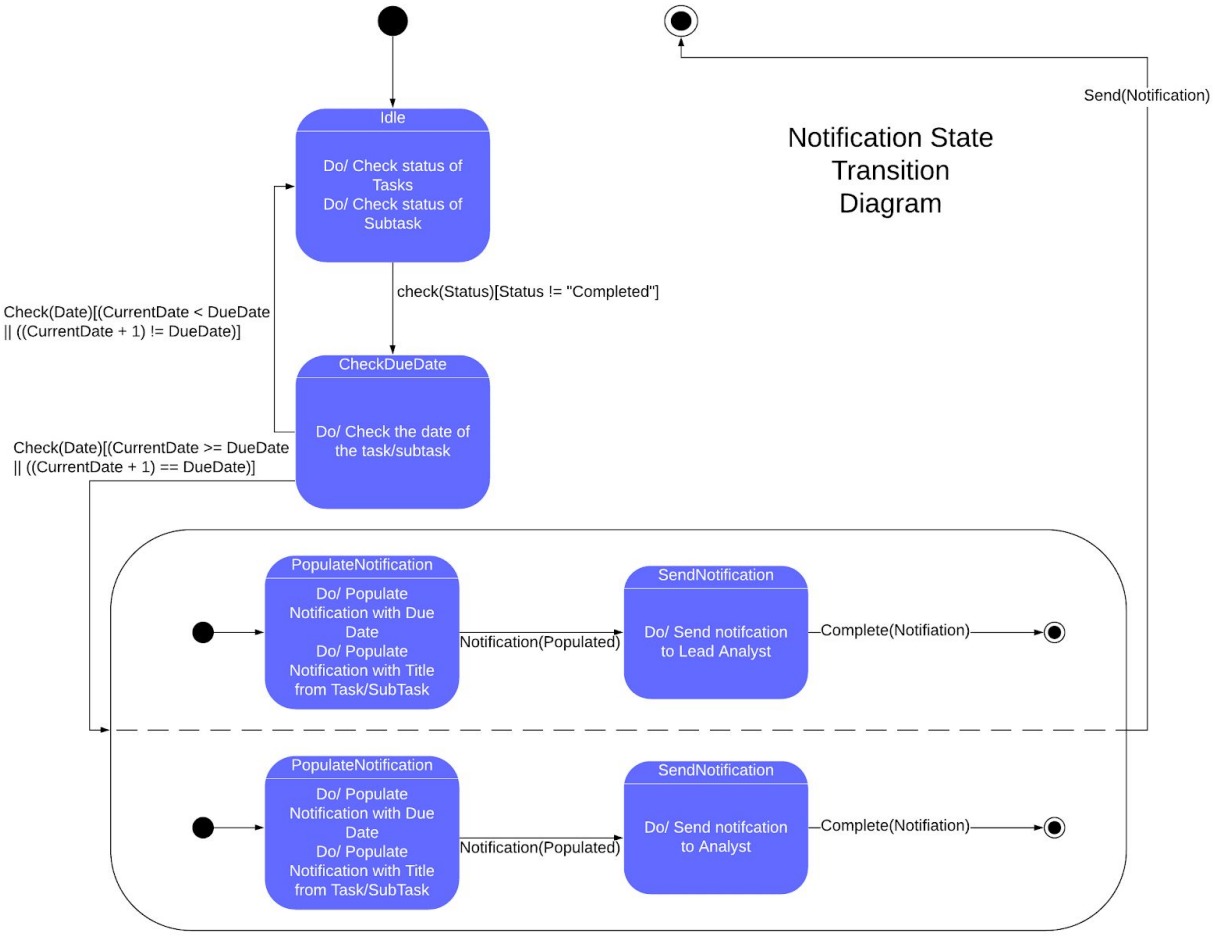
4.1 Finding State Transition Diagram

A state transition diagram is used to represent a finite state machine. This state transition diagram depicts the steps that are necessary to convey the overall view for Findings. The following diagram below shows the clients needs, and how the states transition depending on the behavior of the event.



4.2. Notification State Transition Diagram

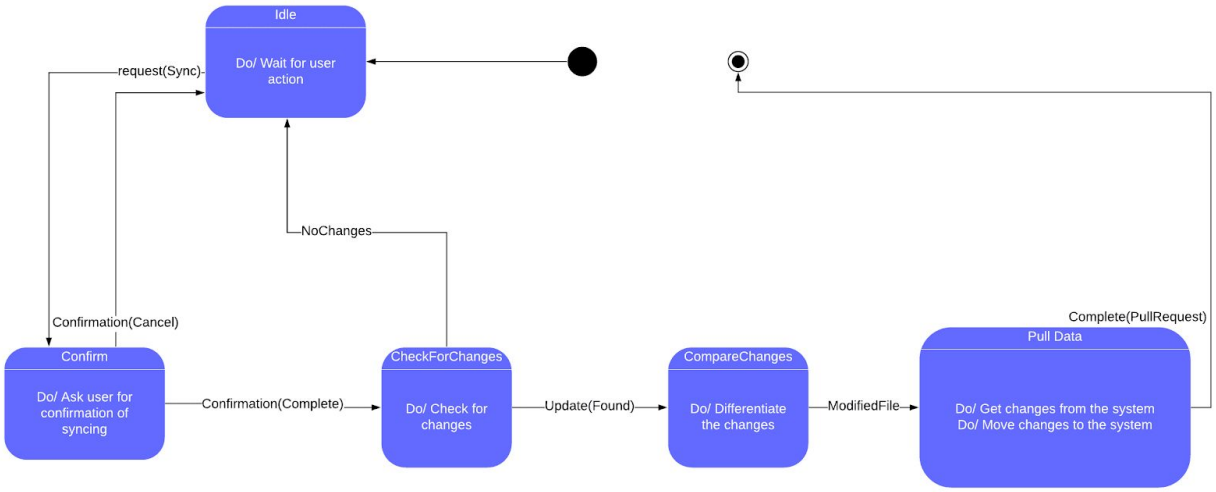
A state transition diagram is used to represent a finite state machine. This state transition diagram depicts the steps that are necessary to convey the overall view for Notifications. The following diagram below shows the clients needs, and how the states transition depending on the behavior of the event.



4.3. Data Synchronization Transition Diagram

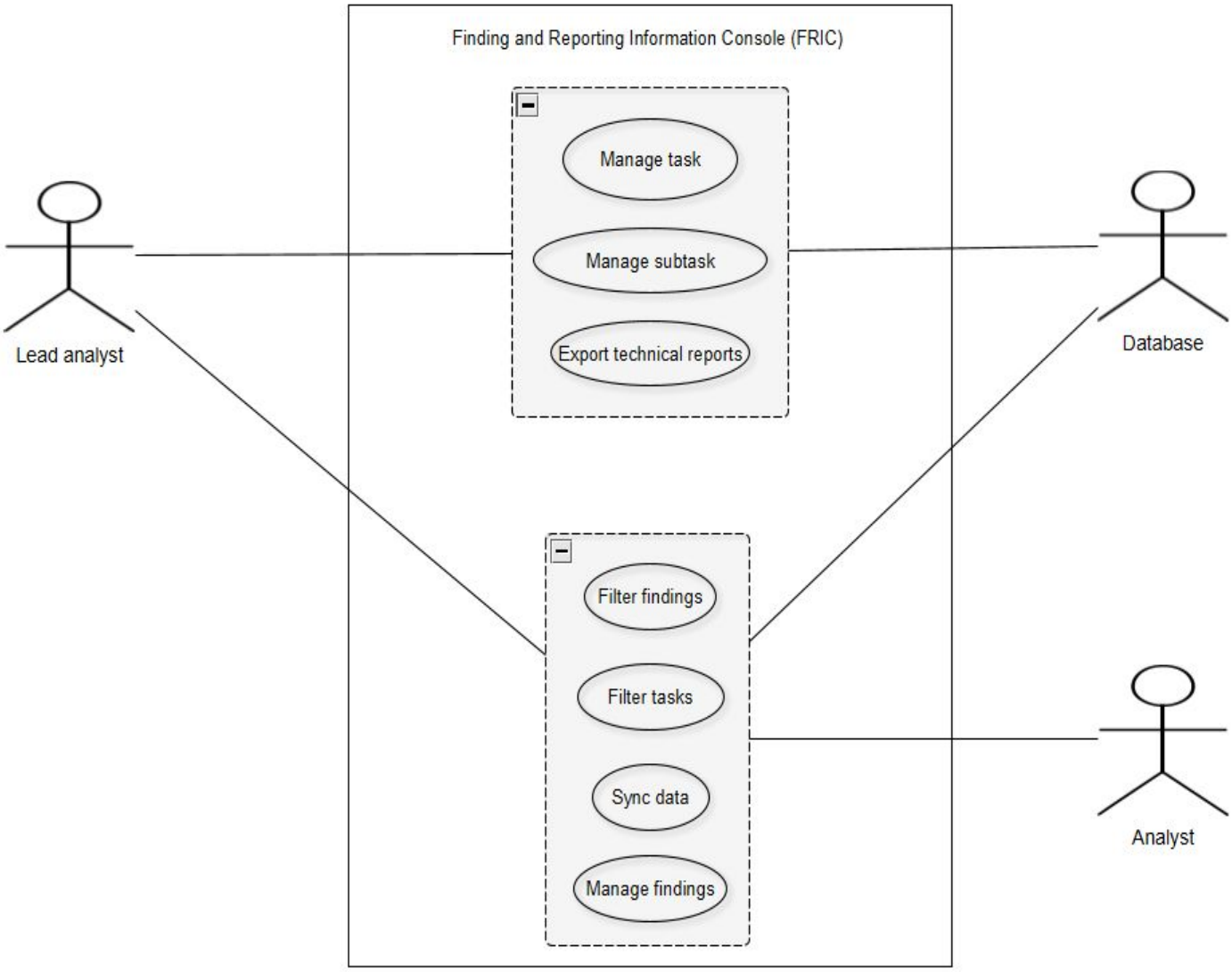
A state transition diagram is used to represent a finite state machine. This state transition diagram depicts the steps that are necessary to convey the overall view for Data Synchronization. The following diagram below shows the clients needs, and how the states transition depending on the behavior of the event.

Sync State Transition Diagram



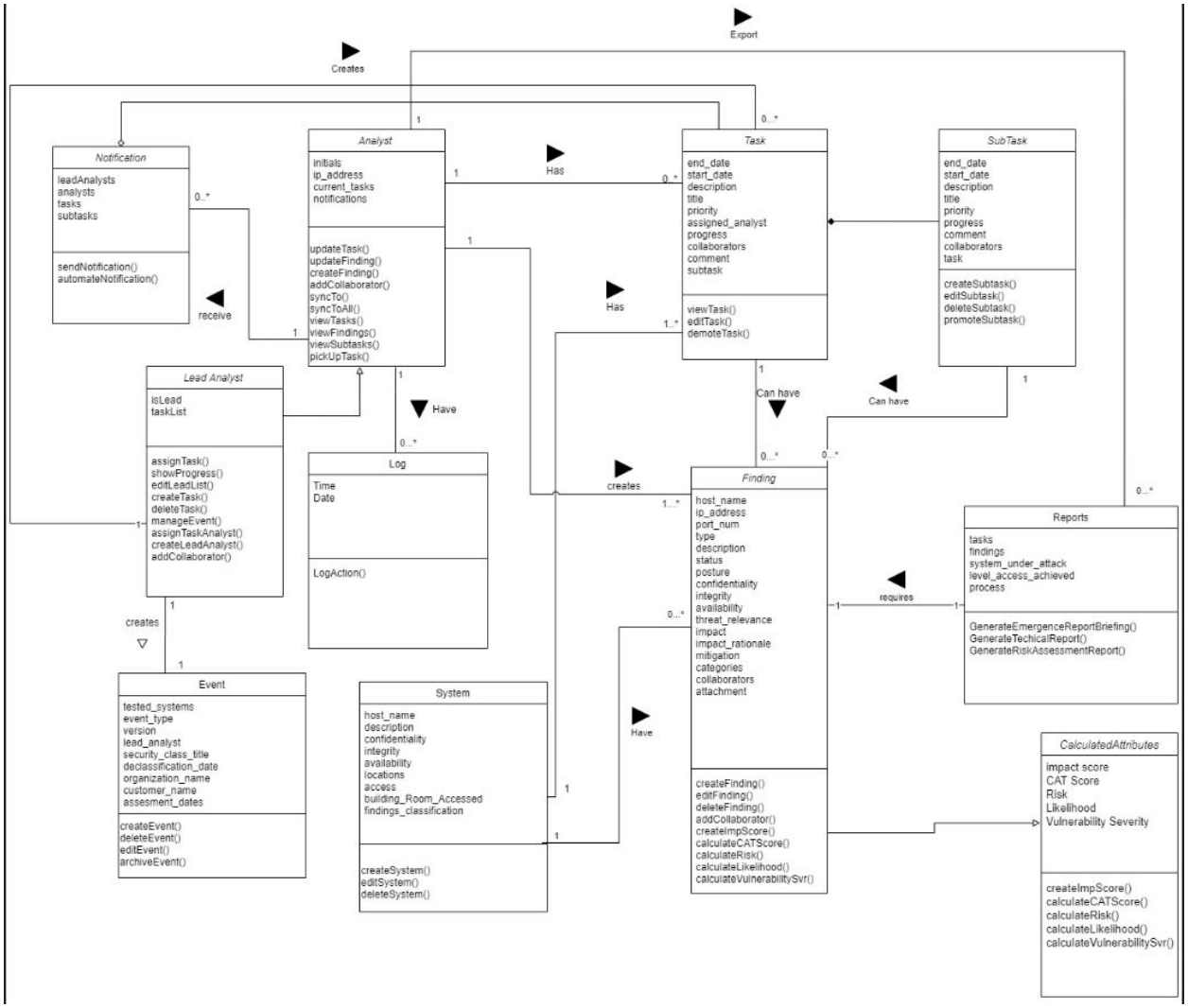
4.4. Use Case Diagram Level 1

The following use case level 1 shows the different actors and their actions within the F.R.I.C system.



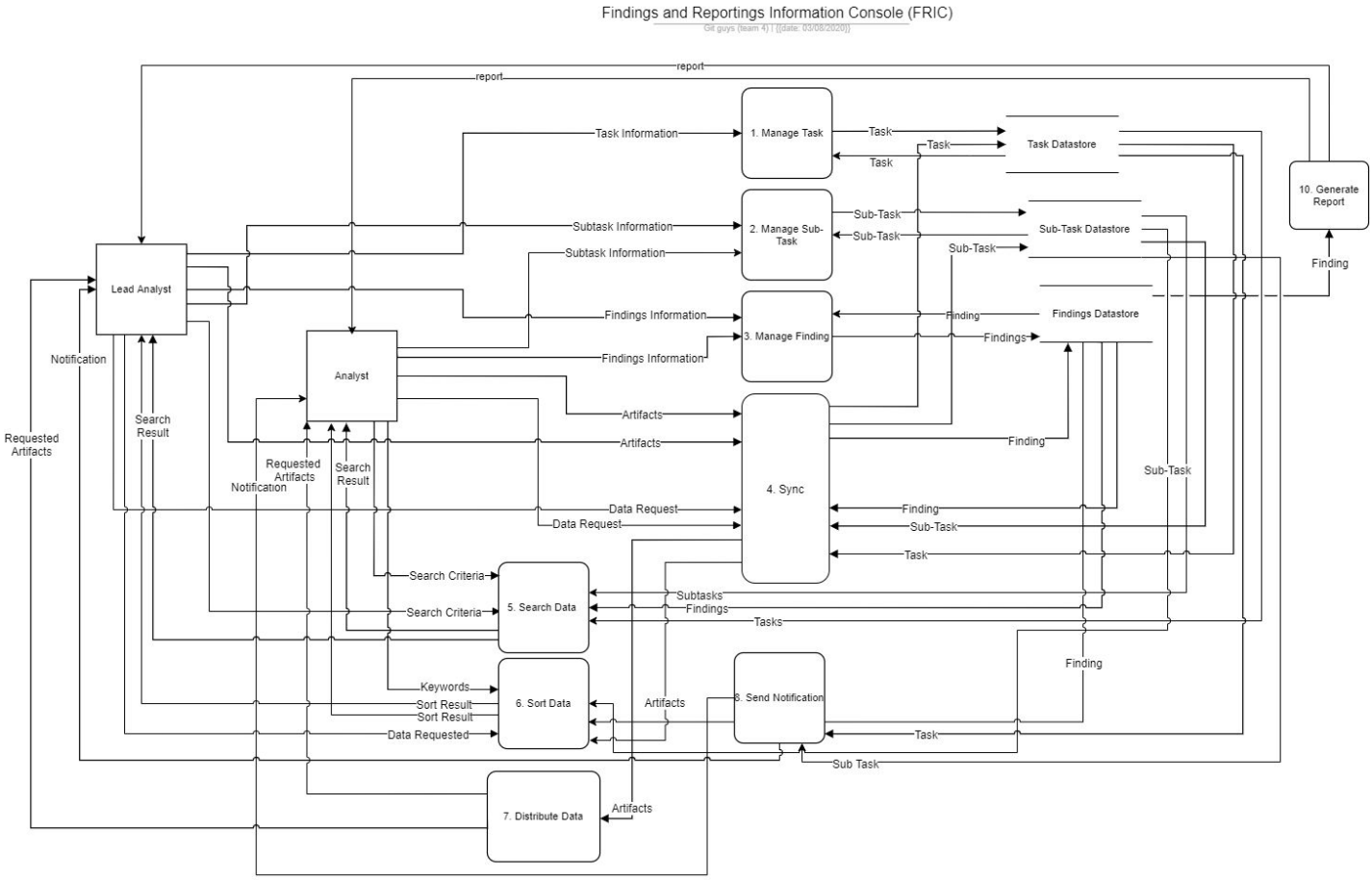
4.5. Class Diagram

The following class diagram represents the static view of the F.R.I.C system showing the attributes, operations, and relationships of the different classes that fulfill the needs of the system.



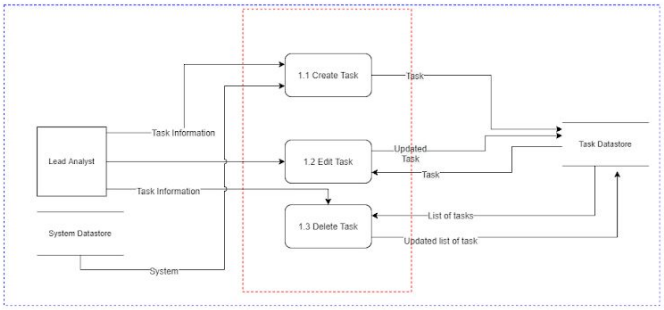
4.6. Data Flow Diagram Level 1

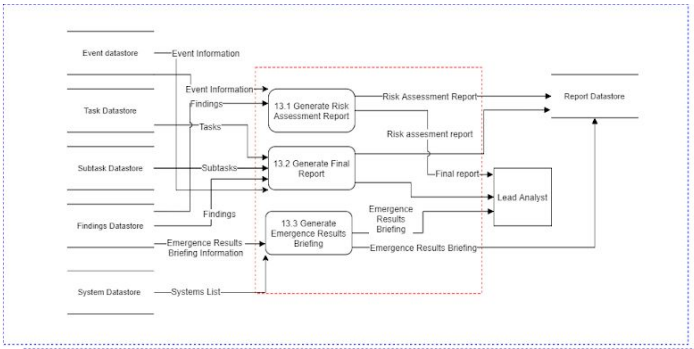
A data flow represents an input/output view of the system and shows the different producers and consumers of data. Each flow of data represents the specific information that will be moving throughout the system while processes, which are represented by rounded rectangles, demonstrate the transformation of data.

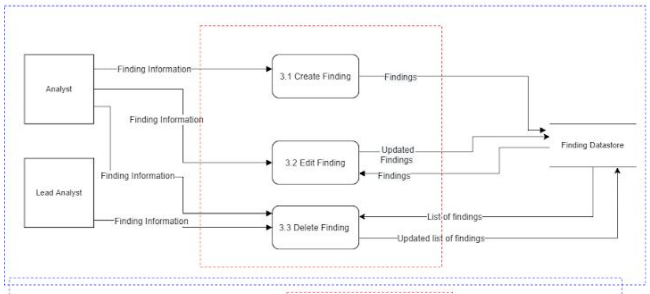


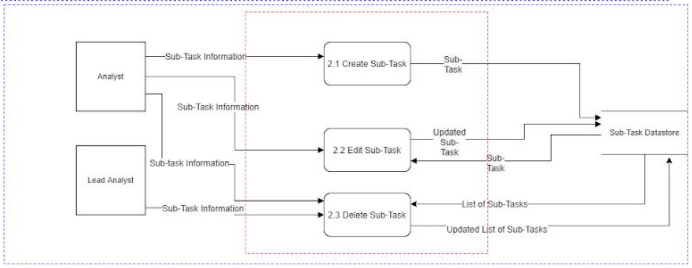
4.7. Data Flow Diagram Level 2

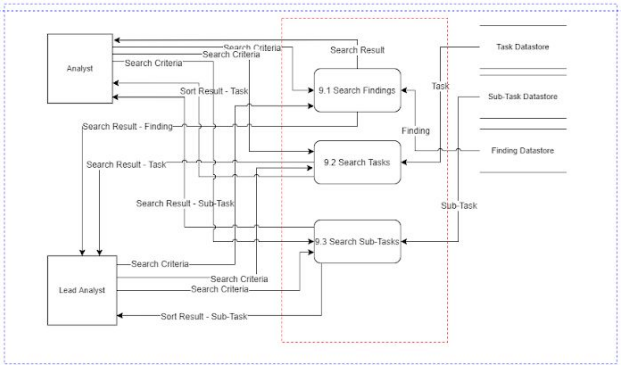
The following level 2 data flow diagrams offer a more detailed look of each one of the processes presented in the level 1 data flow diagram of F.R.I.C.

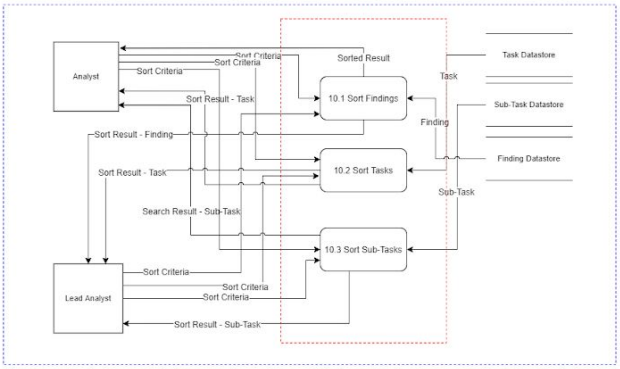












</3