Mitigation Repository

Implementation Document

Prepared by Kristen Stansfield
Organization: River Otters
Date: 5/3/2020
Version: 1

Table of Contents

1 Introduction

1.1 Purpose

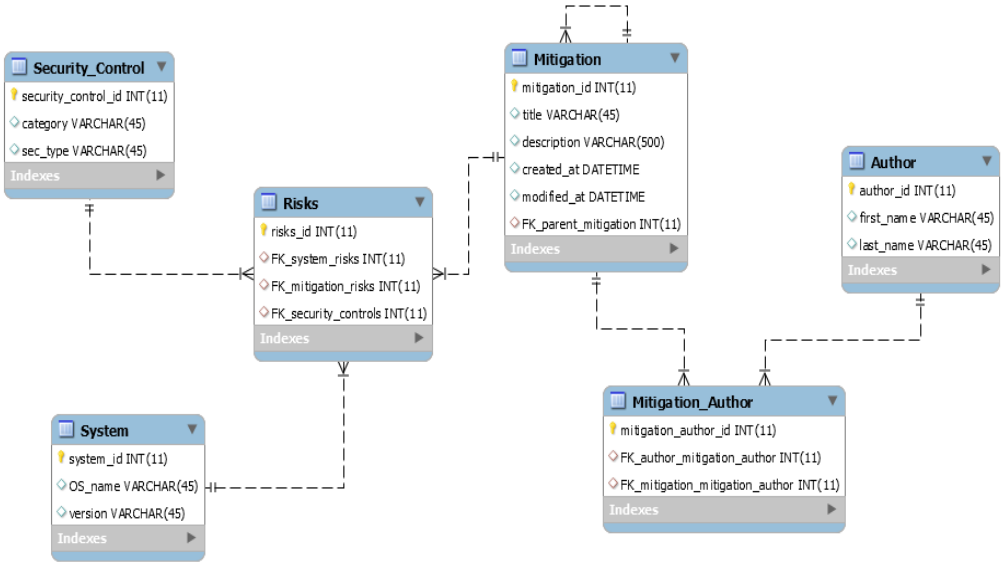
The purpose of this document is to outline the implementation of the Mitigation's Repository Project, organized by the River Otters. We as a team would like to provide the stakeholders with confidence that our project accomplishes all of the requirements, as outlined in the Requirements Document, and explain how our software can be implemented for real-world use at Lockheed Martin. This document will describe important information regarding the system that the future software engineers will need to know in order to implement the project into their own workplace.

1.2 System Overview

The Mitigation Repository will allow the user to perform searches, add, delete, and edit mitigations in the Mitigation Repository. This includes all information related to a mitigation including the author, system, and the security controls. The system allows the user two methods to perform searches to display mitigations to the risks the user is trying to solve. The user can search for mitigations by entering the security controls consisting of the category and type of the risk. If the user does not know the security controls the user can search by the title of the mitigation. The user then is brought to a search results page that shows the mitigations corresponding to their search. The search results page is split in two sides to help the user read the information. The left side of the search results page shows the brief view of the mitigation's that correspond to the user's search. This includes the title and part of the description. When a user clicks on the mitigation it brings the mitigation up on the right hand of the screen in a detailed view. This view shows the title, description, system name, version, and the security controls of the mitigation the user clicked on. The system also allows the user to create a new mitigation from the home page where the user can click on the create new mitigation button which takes the user to the create new mitigation page. In this page the user can enter the required information to create the mitigation. The required information to form a mitigation is the title and description of the mitigation. The security controls the mitigation corresponds too and the system name and version and author of the mitigation

1.2.1 System Description

The Mitigation Repository system contains a User Interface web application that runs alongside a MySQL database. The user, depending on the certain permissions that they have, are able to search, fork, edit, and delete mitigations. All of the information regarding the users, mitigation information, security controls, and system information is stored in the Mitigation Repository Database, and the processes that the project supports are accomplished through database procedures. The Mitigation Repository database contains 4 entities: the mitigation, author, security control, and system. The author contains the author_id, first_name, and last_name attributes, with author_id being its primary key. The Mitigation entity has a title, parent_mitigation, description, created_at, as well as mitigation_id being its primary key. Security_Controls contain a category, a type, and a security_control_id. Database procedures were created to accomplish tasks such as adding, forking, editing, and deleting. These procedures are imperative for the system, as they allow for the ease of implementing these processes. In order to add, edit, and fork a mitigation, the system takes in a Title, Description, OS, OS Version, Category, Type, firstName, and lastName. All of these parameters are of type string.



Mitigation Repository Database Physical Model.

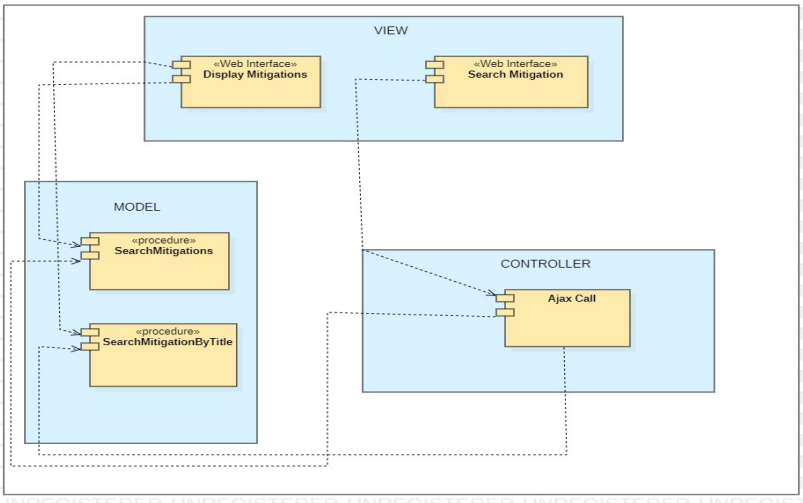
1.2.2 Assumptions and Constraints

The following assumptions are made in planning the implementation of the System:

- The client SE is knowledgeable about using an Amazon Web Server, or alternatively have their own server to host the application.
- The client SE must have the funds to keep the Amazon Web Sever running (Roughly $3 per month), or alternatively have the funds to host their own server.
- The client SE in charge of installation must understand the issues around connecting to a MySQL database.
- The client SE in charge of installation must have all the privileges/username and passwords needed to install and configure the Application Server and the database.

1.2.3 System Organization

The Mitigation Repository system is organized in a Model-View-Controller design. The view is representative of the Web Interface the user will interact with. The controller as labeled represents Ajax Calls which are used by in the system to make asynchronous calls so to the user the view looks like it never goes to another page. This Ajax call is a call to a database procedure. This system uses many calls to different database procedures. This is down to limit the php workload and to simplify the process. The model represents database procedures get information from the database to be displayed onto the view.



1.3 Glossary

This subsection of the Project Implementation Plan lists all terms and abbreviations used in this document.

Forking: the ability to clone a mitigation so users can work on mitigations to add different solutions to the same mitigation.

Repository: a central table for all storage locations. It is used to implement version control and can store multiple versions of a data record. There are countless types of mitigations in cyber security so using centralized data, it ensures that all the data is being pulled and stored in one location.

Risk Mitigation: a strategy to prepare or decrease the presence of threats faced by data center. Usually illustrated in steps for a user to follow to reduce possible negative effects on a system's data center.

Security Control: a way to reduce or mitigate a risk to assets (ex: physical property, information data, hardware systems). These are classified by multiple criteria which is given a category (ex: directive, preventative, detective, etc.) and a type (ex: physical, administrative, or technical)

2 Management Overview

2.1 Description of Implementation

We intend to deploy the implementation of the Mitigation Repository dependent on our stakeholders' time. As the system is fairly simple and does not include too many different hardware and software systems, it could be argued that an "instant-on" approach of implementation for this project would be appropriate. If the stakeholders believe that more time would be needed to implement the system, then the project team would accept those circumstances.

2.2 Points-of-Contact

Role	Name	Contact Email
Scrum Master	Alyssa Indriso	indrisoa5@students.rowan.edu
Product Owner	Anthony Tesoriero	tesoriera6@students.rowan.edu
Development Team	Michael Burke	burkem25@students.rowan.edu
Development Team	David Glennan	glenna29@students.rowan.edu
Development Team	Theresa Morris	morris85@students.rowan.edu
Development Team	Kristen Stansfield	stansfiek0@students.rowan.edu

Table 2.2 – Points-of-Contact

2.3 Major Tasks

2.3.1 Connect to the Web Server

The Mitigation Repository is a web-based application that requires a server to function properly. Throughout the project development, the members of the River Otters relied on an Amazon Web Server to display the working page. However, software engineers may use any web server that they deem fit for this project. They just must adjust the code accordingly. Software Engineers can use any software tool used to transfer files to the internet. To properly implement the Web Server into the project, the software engineer must already have all of the files of the source code on their computers. From there, they can use the file transferring software to connect the web

server to the page. Any software engineer with the intention of changing any of the source code will be responsible for using the corresponding web server for the project, as they may want to access the code and change anything that they see fit to the web page. The acceptance criteria for this specific task is successfully connecting to the web server so the web page correctly displays on any browser. If a software engineer changes any of the code in the project, the web page will successfully update.

2.3.2 Connect to the Database

Software engineers with the development role will be responsible for this task, as they may need to access the database for the project. The Mitigation Repository database is a MySQL based database, so software engineers who are completing this task should be familiar with the workings of MySQL. In order to connect to the database, the staff must have a software that allows them to directly access and manipulate the contents. In the River Otters' case, DataGrip v2019.3 was used in writing Database procedures and general database overview. This tool was chosen due to familiarity with JetBrains IDE's and ease of use on setting up connection to Database. However, software engineers may choose whatever software they are most comfortable with. From there, staff should use the correct Proxy host, user, and private key file that correctly connects to the database. Once this task is completed, the software engineers working with development roles should successfully be able to access and manipulate the database in any way they see fit.

2.3.3 Establish User Roles

Before the implementation of the Mitigation Repository project, staff must establish the roles that they will be responsible for. In the system, there are three roles that a user may have: admin, general user, and development. Anybody who has access to the system may create a general user account for themselves, however admin and development roles are protected by specific criteria for their role. An admin account has a certain username and password just for admins, and development users cannot be created on the web-level. Once a development user is created, they are given the same access as the admin accounts. Before implementation, it is important to establish who will have access to the admin responsibilities and who will have access to the development responsibilities.

2.3.4 Ensure Account Creation/Login Abilities

The Mitigation Repository project relies on a login system to ensure that roles are given the correct permissions based on who is logged in. This system is extremely important for security of the system and to ensure that the system is kept maintained. To create an account, the user can direct themselves to the "Login" page and may enter any credentials about their account on the right-hand side of the page. Upon pressing "Create Account" their information is stored in a database and they will be able to login with those same credentials the next time they would like to log in to the system. Before implementation of the system, all admins, development team members, and a select group of general users should ensure that they can both create an account and login to their account correctly, without error.

2.5 Security and Privacy

The Mitigation Repository product is secured with a login system to ensure that those without admin and development privileges cannot access certain information. In order to ensure that nobody is given access to admin responsibilities without the proper credentials, nobody shall be given the username or password for admin functionality. General users have the ability to create their own account with a username and a password, which will be stored into database columns accordingly. General users do not have the ability to delete or edit any mitigations in the system, only those who are logged into the admin account have those capabilities. General user usernames and passwords are stored in the Mitigation Repository database that is secured by SQL injection prevention, so nobody can get the information about a current user and use their account for their own doing.

2.5.1 System Security Features

To ensure the reliability of the system security, SQL injection is currently prevented in the system. That way, any malicious actors using an SQL injection technique to retrieve any hidden data, such as usernames and passwords of the users, will be stopped. When implementing the Mitigation Repository, we must ensure that SQL injection is still prevented in the system.

3 Implementation Support

3.1 Hardware, Software

This section outlines the proposed hardware and software for implementing the Mitigation Repository system. Client will provide and set up, with assistance from consultant personnel, all hardware and software required for the implementation activities.

3.1.1 Hardware

The hardware that is required for installing and testing this project includes:

- Web Server – the stakeholders must connect to a Web Server in order to use the Mitigation Repository software.
- Computer – the Mitigation Repository is a web-based system, which requires computer and internet access to use.

3.1.2 Software

The software that is required to support the implementation of the Mitigation Repository includes:

- Mitigation Repository Database - The Mitigation Repository Database contains information about the numerous fields having to do with mitigations entered, as well as information about user accounts.
- File Transfer software – Any software that allows software engineers to transfer files to the web server.
- Web Browser – The Mitigation Repository system is compatible with any web browser, including but not limited to Google Chrome, Mozilla Firefox, Safari, and Opera.
- Operating System – The Mitigation Repository system is compatible with any operating system that the software engineers may want to use. This is including but not limited to Windows 10, Windows 7, Mac, and Linux machines.

3.3 Personnel

The Mitigation Repository has three different types of user roles: general user, admin, and development. Therefore, it is important to have individuals to fill all three of those roles to ensure a successful system.

Trusted higher-ups of the company would have the permissions of the admins. These individuals would need to be trained on how to correctly and successfully edit and delete mitigations in the system, as well as be trained to know when it would be appropriate to edit or delete a mitigation. Software engineers who are responsible for regularly updating the system would have development roles. Development roles are no different than an admin role in terms of web-based access, however they have access to all of the source code of the project. These individuals would need the most thorough training about the system, as they would be working directly with the code. Their training would include the ability to connect to the Mitigation Repository database, connecting the web server to the web page, and how to generally maintain the project.

3.3.1 Staffing Requirements

The Mitigation Repository will need a fairly small group of personnel to be staffed for maintenance and admin responsibilities. In the River Otters' case, four members of the development team were responsible for overall maintenance and development of the project. Therefore, once the project is implemented in the workplace a small group of five to ten software engineers should be responsible for keeping the system clean and operationally performing maintenance. Software engineers who are responsible for regularly updating the system would have development roles. These individuals would need the most thorough training about the system, as they would be working directly with the code. They will be required to be highly skilled in working in both front end and back end applications, specifically working in Ajax to maintain asynchronous web applications, Php, jQuery, and MySQL. They will also be responsible for overseeing HTML and CSS code to ensure that the web page stays organized and appealing to all users. The admin team for the Mitigation Repository should also be fairly small, a group of two or three trusted individuals in higher-up positions would be appropriate. The admins of the system do not have any access to the source code of the system and will not be manipulating anything having to do with the system software. Instead, they have the ability to edit and delete mitigations from the system. Therefore, they must be trained in how to perform those tasks as well as the appropriate time to edit or delete a mitigation.

3.4 Outstanding Issues

As of the current time that this Implementation Document is being written, the state of New Jersey is under social distancing orders due to the Covid-19 virus. The majority of people in the workplace, as well as the River Otters, are working remotely, which has the potential to set back the implementation process of the Mitigation Repository. This limits the ability to meet in a face-to-face manner for the future implementation, which would be seemingly less effective than conducting the implementation over a meeting software such as GoToMeeting. However, if come time to implement the system it is agreed upon that the task can be done remotely, this issue is no longer as outstanding.

3.5 Implementation Impact

The implementation of the Mitigation Repository software will give users a central repository to create, organize, and search for mitigations. Akin to a GitHub for mitigations, this software is beneficial to users as it will allow them to view mitigations in one location. Users can create mitigations if they think of one that is not already in the repository, and users are also able to fork off of a mitigation already existing in the database, allowing them to change a certain element of the mitigation to make it their own. Forking the mitigation adds the new information to the database, allowing for other uses to see the newly forked mitigation. This project uses both user and admin control since database monitoring will be necessary to ensure information is the most accurate.

3.6 Performance Monitoring

The performance of the system should be regularly monitored by the development software engineers in order to determine if the implementation was successful. Upon implementation, users should be successfully be able to add, fork, and search for mitigations, and admins should be able to add, fork, search for, delete, and edit mitigations.

4 Communication

4.1 Change Management

It can be ensured that the Mitigation Repository will be accepted by the user community and incorporated into day-to-day activities due to its natural simplicity and user-friendly design. Having a central repository for mitigations will be a key feature in the organization for cyber security professionals. Instead of having to search countless other places for the exact information that they are looking for, the Mitigation Repository system will allow professionals to search in one place, while being able to filter their search even further.

4.2 User Training

General users of the Mitigation Repository system will be trained to interact with the system in the way that it was intended. They will first be taught how to make an account, then how to log into said account. From there, they will be shown a demonstration on how to create a mitigation, thus adding it to the database of other mitigations. Then, once the user would like to search for a mitigation, their mitigation will be displayed there. Users will then be shown how to fork a mitigation, which relies on making the previously establish mitigation into your own. Users will be shown that they can filter their search results depending on specifically what they are looking for in the repository. These can be short training sessions, as there is no complex knowledge needed to be taught to a general user.

4.3 Communication Plan

In order to provide ongoing status updates and announcements to the user community and keep in contact with the stakeholders, the River Otters intend to utilize email and other online practices. Throughout our roughly four-month project, we have communicated with our sponsors through email weekly. We intend to keep the same level of communication if needed as the project progresses in implementation.

Revision History

Version	Date	Updated By	Comments
1.0	05/1/20	Kristen Stansfield	Initial document creation