Cyber Insurance Application

Functional Specification

Dermot Berry

South East Technological University

17/04/2023

Supervisor Dr Christopher Staff

1 Table of Contents

1. Introduction 1

Purpose of a Functional Spec document 1

Project Scope 1

Related Documents 1

Risks Involved 1

System Overview 2

Context Diagram 2

Use/Misuse Case Diagram 3

System Actors 3

User Roles/Requirements 3

Functional Specifications 4

Non-Functional Requirements 4

Use cases 5

Login Description 5

Login 5

Undergo Assessment 5

Completion/Score 6

Logout 6

User Interface 7

Login Interface 7

Login Page 8

Assessment Page 9

Score/Recommendations 10

System Configurations 11

Software Requirements 11

Project Plan 11

Scope 11

Objectives 11

Goals 12

Metrics 12

Schedule 13

2. References 15

1. Introduction

An application which determines the IT Security posture of a third-party vendor. The purpose of the application is to determine areas of a third-party that may be well secure in certain aspects while possibly lacking substantially in another area. The application will select controls from a range of different Frameworks, in particular; NIST and the ISO 27000 series. The application will be a web based, providing scalability and portability to all vendors.

Purpose of a Functional Spec document

Rosencrance (2019) states that "A functional specification is a formal document used to describe a product's intended capabilities, appearance, and interactions with users in detail for software developers. The functional specification is a kind of guideline and continuing reference point as the developers write the programming code."

Commented [CS1]: (author, year) is grand if following a quotation or claim, but if it's used in the way you've used it then we write Author (year).

Project Scope

Implementing a web developed application to provide a user-friendly experience for a potential client seeking Cyber Insurance.

Related Documents

Document Name	***Availability***	***Description***
Research Manual	https://tinyurl.com/3fazb8mz	Assessment Research

Risks Involved

There are several risks involved with the application, threat actors may penetrate the application to reveal security posture of clients gaining valuable intel of areas that could expose weakened areas of the client's security posture. Another Risk would be that a client may enter information incorrectly to falsely secure a higher security rating to decrease reduce its risk from the perspective of the insurer, allowing for a cheaper policy.

System Overview

As the application will reveal the current security posture of a potential client to allow a insurer to decide whether deem a potential clients risk as profitable or not. When the user completes the assessment, it will reveal areas of their business that may be exposed by an incident or accident.

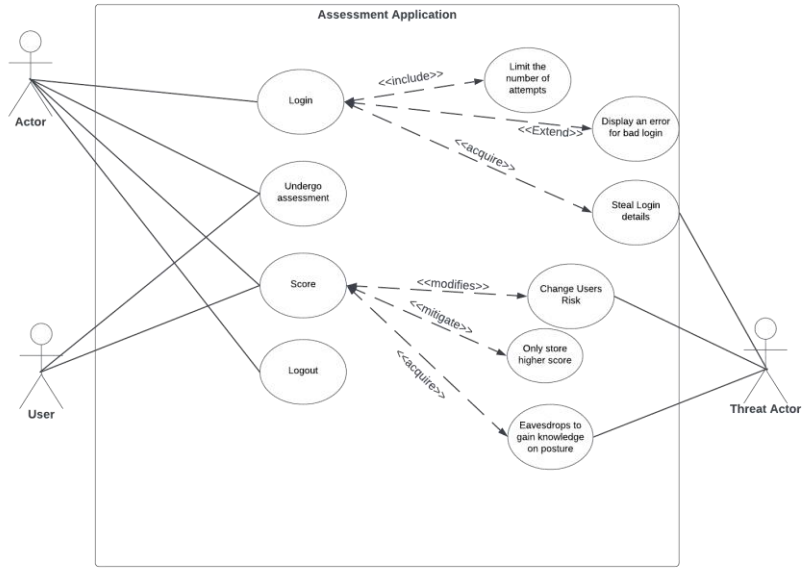
Depending on the outcome of the assessment the Insurance company can determine whether the potential client's risk is too great.

Context Diagram

Pedriquez (2022) states that a context diagram is a high-level representation of the software application to ensure it is easy to understand to all stakeholders involved.

Use/Misuse Case Diagram

According to Olzak (2021) “UML misuse diagrams are often better tools for showing how a system is supposed to behave and how internal and external actors might compromise it.”



System Actors

User Roles/Requirements

User Type	Example	Use Frequency	Access/Features Used	Notes
Actor	Provides data	Frequent	Login, Undergo Assessment, View Score Logout	None
Threat Actor	Is a threat to the Application	Rare	Takes advantage of features	None

User	Receives score after assessment	Occasional/Often	Receives a score of Security Posture	None

Functional Specifications

There are several aspects of the application depending on the type of service, whether they are core or non-core.

Name	Classification
Login	Core
Undergo Assessment	Core
View/Save Score	Core
Logout	Core
Provide Feedback to the Vendor	Non-Core
Provide Feedback to the User	Non-Core
Hash Passwords	Core
No Special Characters on passwords	Core
Failed Login attempts	Core
Idle Timeout	Core

Non-Functional Requirements

Task	Description
Usability	Application will be user friendly
Availability	Application will be available at least 90% of the time
Reliability	Application will be able to carry out the intended task
Security	Ensure user details are encrypted and data cannot be accessed by an unauthorised user
Performance	Application shall run without delay
Scalability	Application will be able to deal with higher traffic volume as it becomes more popular
Maintainability	Changes can be easily made in the future

Use cases

The system should be available to all users of the application.
All uses cases should have appropriate permissions to perform tasks of their clearance.
All users should be authenticated with the system.

Login Description

This is the entry point to the application; this function is required as it differentiates the users and allows them to access other core functions.

Login

UC-1	Login
Primary Actor(s)	Actor
Trigger	Successful login
Pre-conditions	Login is granted
Post-conditions	Users can undertake application, View score, Provide the user with feedback
Main Success Scenario	User takes assessment and the vendor can assess
Extensions	N/A
Priority	High

Undergo Assessment

After the user has logged in and selected “Begin”

UC-2	Take Assessment
Primary Actor(s)	User, Threat Actor
Trigger	After login user will be prompted to take an assessment or view the score
Pre-conditions	The user has navigated to the assessment interface. The user confirms each selection for each question
Post-conditions	The user completes the assessment, and the score is stored

	Depending on the result, recommendations will be provided to the user
Main Success Scenario	User takes the assessment and gains a score for their security posture.
Extensions	If the user clicks “Back” The user will return to the previous question.
Priority	High

Completion/Score

User has finished the assessment, the score will be informed to the user, and possible recommendations sent to the user.

UC-3	Score/Recommendations
Primary Actor(s)	User
Trigger	User initiates “Complete” on form
Pre-conditions	User has undergone many questions on the form
Post-conditions	User is given a score User can see recommendations User can retake assessment
Main Success Scenario	Assessment is analysed and user is informed of the score and recommendations to improve
Extensions	Not applicable
Priority	High

Logout

After the user has completed the assessment, obtained their score and possible recommendations received.

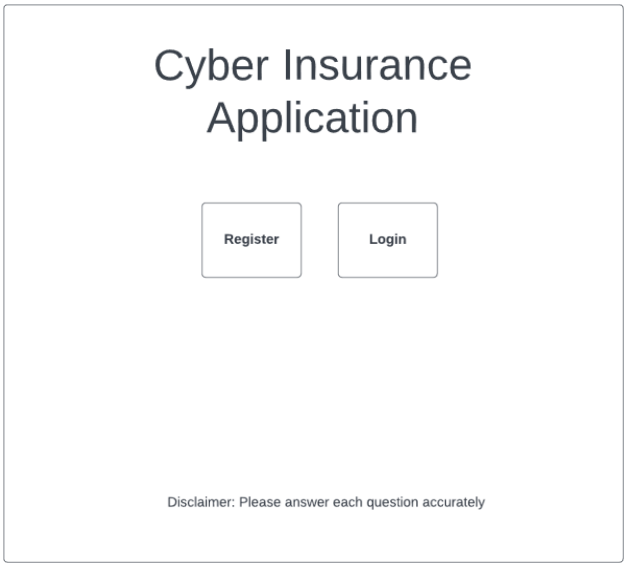
UC-4	Logout
Primary Actor(s)	User
Trigger	User initiates “logout”
Pre-conditions	User has undergone many questions on the form
Post-conditions	User is given a score User can see recommendations User can retake the assessment
Main Success Scenario	Assessment is analysed, and the user is informed of the score and recommendations to improve
Extensions	Not applicable
Priority	High

User Interface

There are several pages that the user will have the ability to navigate through including the Login, Registration, Assessment, and Score page and Logout.

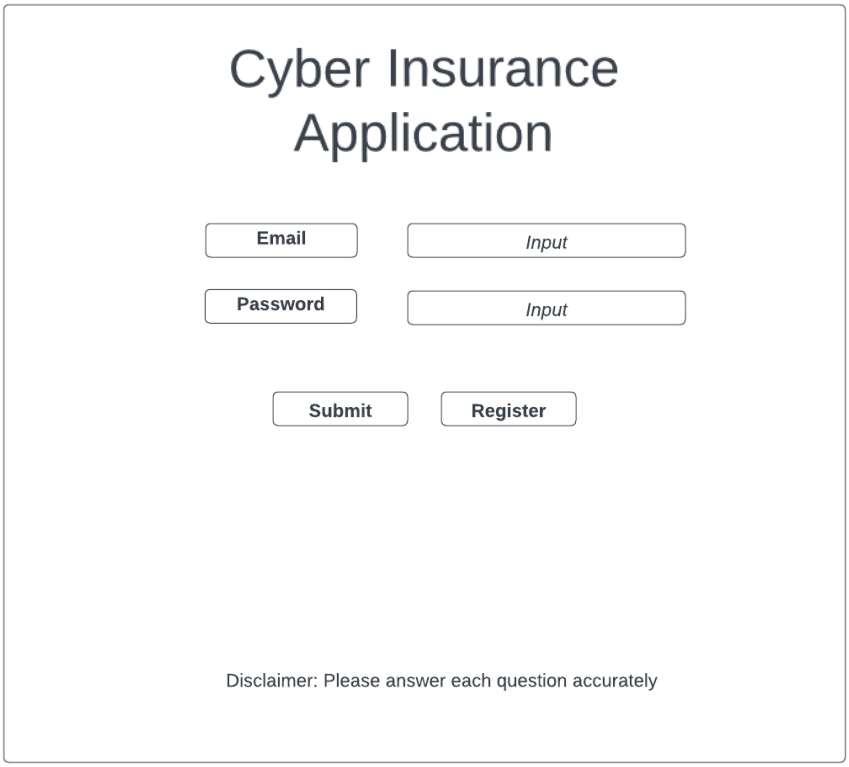
Login Interface

Below is an approximate vision of the initial Login page.



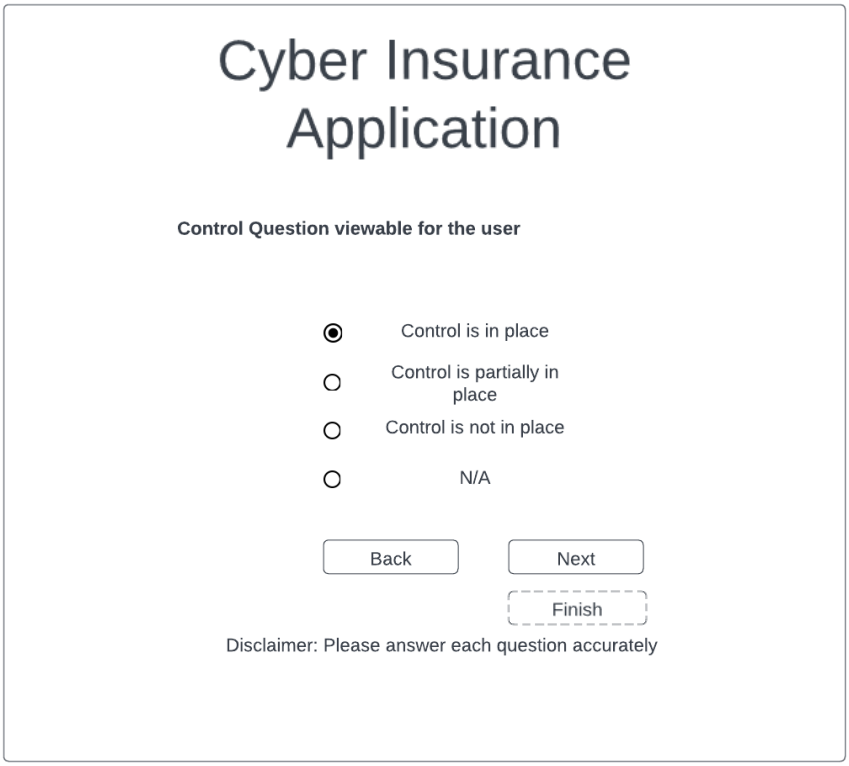
Login Page

After the user has clicked the “Login” button on the homepage it will open the Login page below.



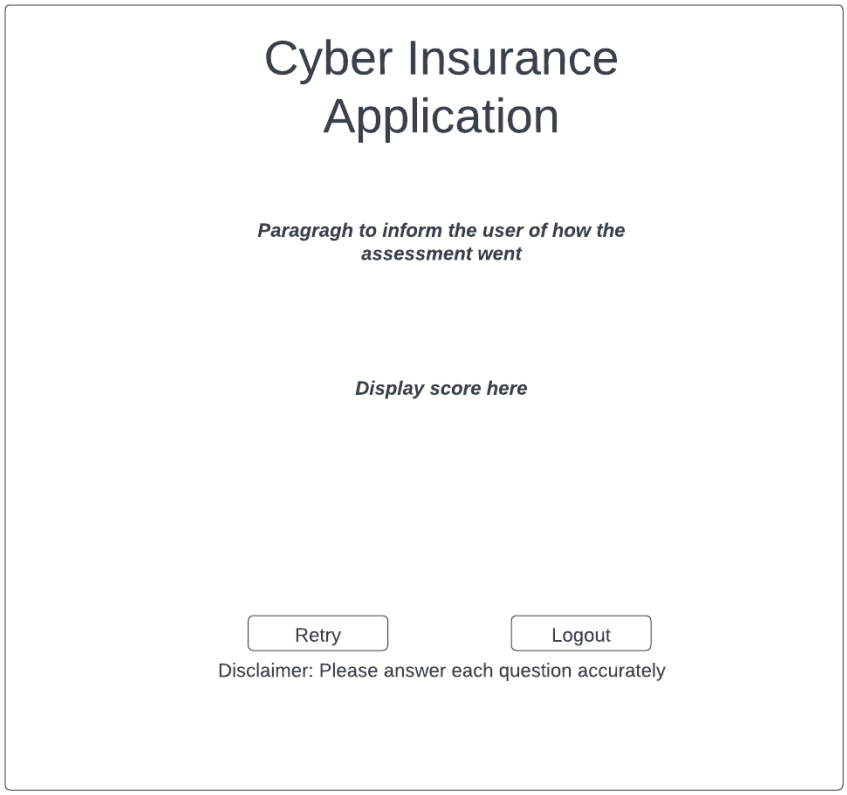
Assessment Page

The assessment page will use a frame that will change its contents after each question is submitted. After the final question has been completed the “Finish” button will display and end the assessment



Score/Recommendations

After the user has taken the assessment, their score will be shown in centre of the page. If there are not enough controls in place, the user will not be able to obtain insurance as their risk will be too high.



System Configurations

A list of steps involved to create a useful system.

Step	Description
1	Configure a web application with a form to ask numerous security questions
2	Setup a database on MySQL with numerous tables with control questions in each table.
3	Store usernames and Passwords which are hashed on MySQL database
4	Store user score from assessment on MySQL database
5	Display score and recommended actions that benefit the user.

Software Requirements

HTML with CSS – Web application and design

PHP - Used to add functionality, to implement the form on the web application.

MySQL – Database to store users, passwords, and questions

Project Plan

To complete the Project within the given time constraints I plan to reach certain goals within the dates shown below. The basics of a project include the following: Scope, objectives, goals, and schedule according to (Lutkevich, 2021).

Scope

With time being a major factor in the research of the project, I will research heavily in one section of the security posture. Each section will be covered substantially to ensure that the necessary research has been carried out in all aspects of the security posture of the third-party.

Objectives

There are several objectives that require a lot of time before advancements can be made. The initial objective is to select controls from each aspect of the IT infrastructure. Controls will be used that are a criticality of 8 or above for the application. Controls will be gathered

from various Frameworks, primarily working with the NIST 800-53 and ISO 27001 Frameworks. To assign criticality each control will have to be weighted by using a DRED matrix. Risk Methodology will be implemented also to display a risk score with each control. Questions for the user will be created from each control. After the controls have been selected the coding section of the project will commence. Initially to set up the database for usernames and passwords. Another database to store the store the questions so that the app form can pull them down one by one.

Goals

The main goals are to have the control questions in place to add to the application.
Secure the application, by hashing, character sanitization, session management etc.
Create a user-friendly application for the user.
Test and document the performance of the app.

Metrics

To determine how successful the application would be, it will depend on the following:

The application will be resistant to SQL attacks.

The application will resist brute force attacks on the registration and login pages.

The application will be resistant to XSS attacks.

The application will be user friendly.

The application will handle each page using sessions.

Pen test will be carried out determine if there any issues found and mitigate them.

The application works on multiple platforms, Chrome, Firefox, Edge etc.

Ensure that the application is reliable and performs as intended.

Schedule

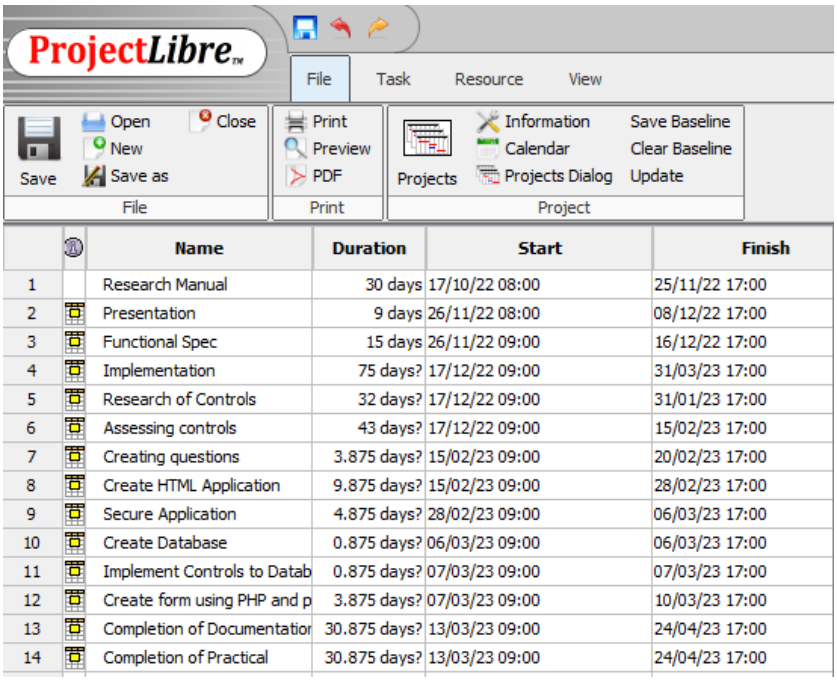
Task	Start Date	End Date	Duration
Research Manual	17/10/2022	25/11/2022	40
Presentation	26/11/2022	08/12/2022	12
Functional Spec	26/11/2022	16/12/2022	20
Implementation	17/12/2022	31/03/2023	105
Research of Controls	17/12/2022	31/01/2023	46
Assessing controls	17/12/2022	15/02/2023	61
Creating questions	15/02/2023	20/02/2023	5
Create HTML Application	15/02/2023	28/02/2023	14
Secure Application	28/02/2023	06/03/2023	7
Create Database	06/03/2023	06/03/2023	1
Move controls to database	07/03/2023	07/03/2023	1
Create form using PHP and pull questions	07/03/2023	10/03/2023	3

1.1.1.1 Project Libre Breakdown of application

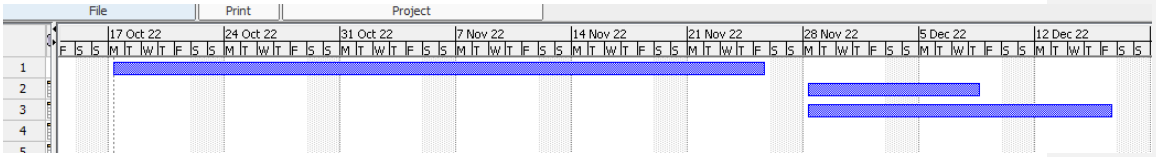
Must be opened using Project Libre

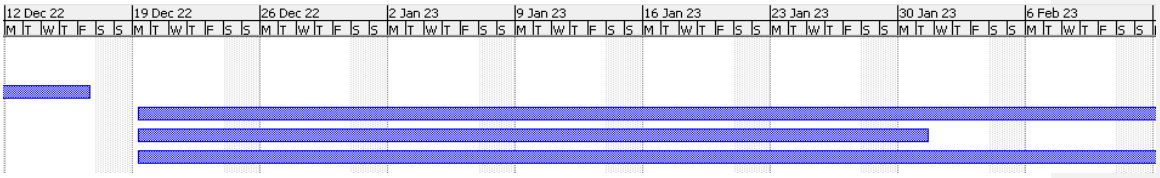
In the event the file below cannot open, see figures below.

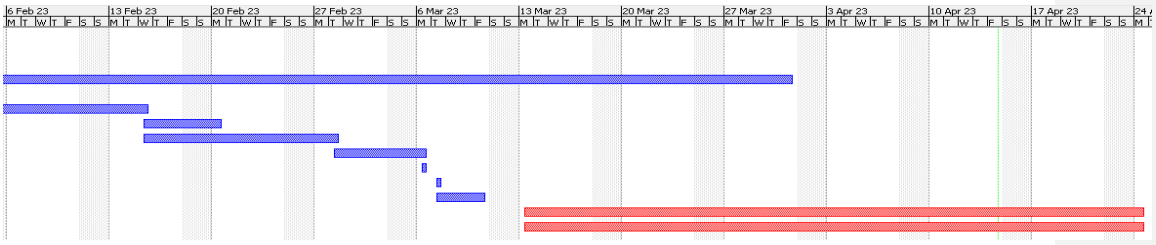
FunctionalSpec.pod



		Name	Duration	Start	Finish
1		Research Manual	30 days	17/10/22 08:00	25/11/22 17:00
2		Presentation	9 days	26/11/22 08:00	08/12/22 17:00
3		Functional Spec	15 days	26/11/22 09:00	16/12/22 17:00
4		Implementation	75 days?	17/12/22 09:00	31/03/23 17:00
5		Research of Controls	32 days?	17/12/22 09:00	31/01/23 17:00
6		Assessing controls	43 days?	17/12/22 09:00	15/02/23 17:00
7		Creating questions	3.875 days?	15/02/23 09:00	20/02/23 17:00
8		Create HTML Application	9.875 days?	15/02/23 09:00	28/02/23 17:00
9		Secure Application	4.875 days?	28/02/23 09:00	06/03/23 17:00
10		Create Database	0.875 days?	06/03/23 09:00	06/03/23 17:00
11		Implement Controls to Datab	0.875 days?	07/03/23 09:00	07/03/23 17:00
12		Create form using PHP and p	3.875 days?	07/03/23 09:00	10/03/23 17:00
13		Completion of Documentatior	30.875 days?	13/03/23 09:00	24/04/23 17:00
14		Completion of Practical	30.875 days?	13/03/23 09:00	24/04/23 17:00







2. References

Rosencrance, L. (2019). *What is a functional specification document? Software Quality*. TechTarget. Available at: https://www.techtarget.com/searchsoftwarequality/definition/functional-specification (Accessed: December 6, 2022).

Pedriquez, D. (2022) *What is a context diagram (and how can you create one), Venngage*. Available at: https://venngage.com/blog/context-diagram/ (Accessed: December 6, 2022).

Olzak, T. (2021) *Using UML misuse diagrams to secure systems from threat actors*, *Spiceworks It Security*. Available at: https://www.spiceworks.com/tech/it-strategy/articles/using-misuse-case-diagrams/ (Accessed: December 6, 2022).

Lutkevich, B. (2021) Project planning: What are it and steps to create a plan: Searchcio, CIO. TechTarget. Available at: https://www.techtarget.com/searchcio/definition/project-planning (Accessed: December 13, 2022).