



MTU

**Web Based Application to Manage
Company Assets and their Associated
Risks**

by

Aphral Griffin

This thesis has been submitted in partial fulfillment for the
degree of Bachelor of Science in Software Development

in the
Faculty of Engineering and Science
Department of Computer Science

May 2022

Declaration of Authorship

I, Aphral Griffin , declare that this thesis titled, Web Based Application to Manage Company Assets and their Associated Risks and the work presented in it are my own. I confirm that:

- This work was done wholly or mainly while in candidature for an undergraduate degree at Munster Technological University.
- Where any part of this thesis has previously been submitted for a degree or any other qualification at Munster Technological University or any other institution, this has been clearly stated.
- Where I have consulted the published work of others, this is always clearly attributed.
- Where I have quoted from the work of others, the source is always given. With the exception of such quotations, this project report is entirely my own work.
- I have acknowledged all main sources of help.
- Where the thesis is based on work done by myself jointly with others, I have made clear exactly what was done by others and what I have contributed myself.

Signed: Aphral Griffin

Date: 03/05/22

Munster Technological University

Abstract

Faculty of Engineering and Science
Department of Computer Science

Bachelor of Science

by Aphral Griffin

This project aims to develop a web-based application that companies can use to manage their assets and the risks associated with those assets. Within this application, companies will be able to enter assets by category, identify the risks to these assets, and then select what controls are being used to help mitigate those risks. There will be a tagging system which will provide templates for companies to use so that they can get set up on the application faster.

Acknowledgements

I would like to say thank you to Cliona McGuane in Munster Technology University for her amazing help and encouragement during the implementation of this project. This project would not be what it is without her advice and guidance. Her support not only with this project but with all my modules this semester has been invaluable and I can not thank you enough.

I would like to thank Dr.Mubashir Rehmani in Munster Technology University for his help during the research phase of this project.

I would also like to thank all the the staff at Munster Technology University as well as all the lecturers in the Computer Science department for all their help over the last 4 years.

Contents

Declaration of Authorship	i
Abstract	ii
Acknowledgements	iii
List of Figures	vii
List of Tables	xi
Abbreviations	xiii
1 Introduction	1
1.1 Motivation	1
1.2 Executive Summary	1
1.3 Contribution	2
1.4 Structure of This Document	3
2 Background	4
2.1 Thematic Area within Computer Science	4
2.2 Project Specifics - Minimal Background Knowledge	4
2.2.1 Web Applications	4
2.2.2 Security Management	4
2.2.3 Artificial Intelligence	5
2.2.4 Assets	5
2.3 Identify Community	5
2.4 A Review of Management of Assets and their Associated Risks	7
2.4.1 A Literature Review of Management of Assets and their Associated Risks	7
2.4.2 Practical Applications Review of Management of Assets and their Associated Risks	13
2.4.3 Review Summary	15
3 Problem - Web based application to manage company assets and their associated risks.	16

3.1	Problem Definition	16
3.2	Objectives	17
3.3	Functional Requirements	18
3.4	Non-Functional Requirements	19
4	Implementation Approach	20
4.1	Architecture	20
4.1.1	Technologies Involved	21
4.1.2	File Structure	23
4.1.3	Database Overview	24
4.2	Use-Case Description	25
4.3	Risk Assessment	38
4.3.1	Risk 1	38
4.3.2	Risk 2	38
4.3.3	Risk 3	38
4.3.4	Risk 4	38
4.4	Methodology	39
4.5	Implementation Plan Schedule	40
4.6	Evaluation	41
4.7	Wire Frames	41
5	Implementation	45
5.1	Difficulties Encountered	45
5.1.1	Difficulty 1	45
5.1.2	Difficulty 2	46
5.1.3	Difficulty 3	46
5.1.4	Difficulty 4	46
5.1.5	Difficulty 5	47
5.1.6	Difficulty 6	47
5.1.7	Difficulty 7	47
5.2	Actual Solution Approach	48
5.2.1	Architecture of the Solution	48
5.2.2	Technologies Used	49
5.2.3	Use Cases	49
5.2.4	Risk Assessment	50
5.2.4.1	Risk 2	50
5.2.4.2	Risk 3	51
5.2.5	Methodology to Develop Solution Approach	52
5.2.6	Implementation Schedule	53
5.2.7	Evaluation	54
5.2.8	Prototype of Resulting Product	55
6	Testing and Evaluation	63
6.1	Use Case Testing - Functional Requirements	63
6.1.1	Use Case 1 - Login	63
6.1.2	Use Case 2 - Maintain Asset	65
6.1.3	Use Case 3 - Maintain Risk	70

6.1.4	Use Case 4 - Maintain Control	75
6.1.5	Use Case 5 - View Assets, Risks & Controls	80
6.1.6	Use Case 8 - Tagging	81
6.2	Portability Testing - Non-Functional Requirements	83
6.2.1	Different Operating Systems	83
6.2.2	Different Devices	84
6.2.3	Different Browsers	85
6.3	Results	88
6.3.1	Use Case Testing Results	88
6.3.2	Portability Testing Results	89
6.3.3	Industry Product Tests	90
7	Discussion and Conclusions	91
7.1	Solution Review	91
7.2	Project Review	92
7.3	Future Work	93
7.4	Conclusion	94
Bibliography		95

List of Figures

4.1	High Level Architecture Diagram for Web Application	20
4.2	High Level Overview of the File Structure for Web Application	23
4.3	High Level Overview of the Database for Web Application	24
4.4	Flow diagram for use case 1	26
4.5	Flow diagram for use case 2	28
4.6	Flow diagram for use case 3	30
4.7	Flow diagram for use case 4	32
4.8	Flow diagram for use case 5	34
4.9	Flow diagram for use case 6	36
4.10	Flow diagram for use case 7	37
4.11	Implementation Plan for Semester 2	40
4.12	Wire Frame for Home Page of the application	42
4.13	Wire Frame for page to display list of all the assets	42
4.14	Wire Frame for Asset Page where detail of one asset is displayed	43
4.15	Wire Frame for Risk Page where detail of one risk is displayed	43
4.16	Wire Frame for Control Page where detail of one control is displayed	44
5.1	Solution High Level Architecture Diagram for Web Application	48
5.2	Trello Board for Sprint	52
5.3	Implementation Chart	53

5.4	Project Backlog	54
5.5	Login Page	55
5.6	Register Details Page	55
5.7	Home age	56
5.8	Asset List Page	56
5.9	Asset Detail Page	57
5.10	Update Asset Page	57
5.11	Risk Detail Page	58
5.12	Update Risk Page	58
5.13	Control List Page	59
5.14	Add Control Page	59
5.15	Control Details Page	60
5.16	Update Control Page	60
5.17	Add Asset Page	61
5.18	Add Asset Page Tags	61
5.19	Add Risk Page	62
5.20	Add Risk Page Tags	62
6.1	User Enters Valid Username and Password and Clicks Login	64
6.2	User on Home Page after Login	64
6.3	Error Message Displayed	64
6.4	User Successfully Adds Asset	66
6.5	Cancel Button Clicked	66
6.6	Error Adding Asset	67
6.7	User Successfully Updates Asset	67
6.8	Cancel Button Clicked Update Asset	67

6.9 Error Updating Asset	68
6.10 Warning Message When Deleting Asset	68
6.11 Successfully Deleted Asset	68
6.12 User Presses Cancel	69
6.13 Error Deleting Asset	69
6.14 User Successfully Adds Risk	71
6.15 Cancel Button Clicked	71
6.16 Error Adding Risk	72
6.17 User Successfully Updates Risk	72
6.18 Cancel Button Clicked Update Risk	72
6.19 Error Updating Risk	73
6.20 Warning Message When Deleting Risk	73
6.21 Successfully Deleted Risk	73
6.22 User Presses Cancel	74
6.23 Error Deleting Risk	74
6.24 User Successfully Adds Control	76
6.25 Cancel Button Clicked	76
6.26 Error Adding Control	77
6.27 User Successfully Updates Control	77
6.28 Cancel Button Clicked Update Control	77
6.29 Error Updating Control	78
6.30 Warning Message When Deleting Control	78
6.31 Successfully Deleted Control	78
6.32 User Presses Cancel	79
6.33 Error Deleting Control	79
6.34 Views Assets	80

6.35 Views Risks	81
6.36 Views Controls	81
6.37 Enters Data No Labels	82
6.38 User Can View Filtered Data	82
6.39 No Relevant Data	82
6.40 Web Application Working on Windows	83
6.41 Web Application Working on Mac	84
6.42 Web Application Working on iPhone	84
6.43 Web Application Working on Android	84
6.44 Web Application Working on Laptop	85
6.45 Web Application Working on Microsoft Edge	85
6.46 Web Application Working on Chrome	86
6.47 Web Application Working on Firefox	86
6.48 Web Application Working on Safari	87

List of Tables

2.1	Small Business Threat Examples	8
4.1	Use Case 1 - Login	25
4.2	Use Case 2 - Add, Edit and Delete Asset	27
4.3	Use Case 3 - Add, Edit and Delete Risk	29
4.4	Use Case 4 - Add, Edit and Delete Control	31
4.5	Use Case 5 - View Assets, Risks and Controls Control	33
4.6	Use Case 6 - Admin Adds Organisation	35
4.7	Use Case 7 - AI Suggests Relevant Documents	37
4.8	Potential Risk 1 - spaCy	38
4.9	Potential Risk 2 - Cloud Functions	38
4.10	Potential Risk 3 - Cloud Firestore	38
4.11	Potential Risk 4 - Google Docs API	38
4.12	Initial risk matrix	39
5.1	Difficulties Encountered Categories	45
5.2	Use Case 8 - Tagging System	50
5.3	Potential Risk 2 - Cloud Functions	50
5.4	Potential Risk 3 - Cloud Firestore	51
6.1	Use Case 1 - Login	63

6.2	Use Case 2 - Maintain Asset - Add Asset	65
6.3	Use Case 2 - Maintain Asset - Update Asset	65
6.4	Use Case 2 - Maintain Asset - Delete Asset	66
6.5	Use Case 3 - Maintain Risk - Add Risk	70
6.6	Use Case 3 - Maintain Risk - Update Risk	70
6.7	Use Case 3 - Maintain Risk - Delete Risk	71
6.8	Use Case 4 - Maintain Control - Add Control	75
6.9	Use Case 4 - Maintain Control - Update Control	75
6.10	Use Case 4 - Maintain Control - Delete Control	76
6.11	Use Case 5 - View Asset	80
6.12	Use Case 5 - View Risk	80
6.13	Use Case 5 - View Control	80
6.14	Use Case 8 - Tagging	81
6.15	Use Case Testing Results	88
6.16	Portability Testing Results	89

Abbreviations

AI	Artificial Intelligence
API	Application Programming Interface
GDPR	General Data Protection Regulation
HSE	Health Service Executive
HTTPS	Hypertext Transfer Protocol Secure
IDE	Integrated Development Environment
ISO	International Organisation of Standards
SDK	Software Development Kit

Dedicated to my friends and family for their endless support and encouragement. This project would not have been completed without all of you.

Chapter 1

Introduction

1.1 Motivation

In recent years companies have been hugely affected by attacks on their assets and though there are many standards to help mitigate these issues, companies have difficulties implementing these standards. To help companies better manage their assets this project aims to make a web-based application that can help companies manage not only their assets but the risks associated with those assets and the controls in place to protect against these risks.

To help smaller companies such as startups and shops, the web application developed in this project will have templates that will allow companies to get set up on the application faster.

Larger companies can have a large number of policy documents that can make it difficult to find the policies that are relevant to the assets. The web application will have an AI to help with this issue, the AI will look at policy documents the company has uploaded and notify the user of policies that may apply to their assets.

1.2 Executive Summary

This project's goal is to help companies better manage their assets and the risks associated with those assets, when this project is complete the aim is to have a web-based application that will help users achieve this goal.

The main feature of the app will be the ability of the user to view all the assets within a company and manage specific assets that are assigned to them. The user will be able

to view all the risks associated with an asset, view the controls on the asset to prevent those risks, view the plan in place if one of those risks occurs, view who is in charge of the asset and any regulations that may apply to that asset.

The application will allow the user to log in to view the assets, when the user first logs in they are brought to a home page that will display notifications about assets they own along with assets that need to be reviewed soon.

The database for this application will contain all the assets and user information. The application will allow the user to create, update and delete assets, risks and controls. This will then be stored in the cloud database.

Users will be able to upload policy and procedure documents to the application, this will be used for an AI which is built into the application. The AI will look at these documents and then suggest policies and procedures that may apply to an asset the user is managing.

Another feature the project aims to have is templates, these templates will help startups and smaller businesses get set up with the application faster. This will also hopefully help smaller businesses who may not have thought about some forms of asset protection or have struggled in the past to know where to begin and can not afford other asset management tools as they can be expensive for smaller businesses.

1.3 Contribution

This project aims to help companies manage their assets, risks associated with those assets and controls in place to help prevent those risks. This will be done by developing a web application for companies to use, this application will allow them to log in and manage their assets.

The web application will be developed using TypeScript[1] and React[2] with MUI[3] for the frontend and Go and Firebase Cloud Functions[4] for the backend. The application will be hosted allowing for it to be accessed by users through any browser.

The web application will also offer templates to help companies get started with managing their assets. These templates will be stored in Cloud Firestore[4] which is the database being used for this project.

There will also be an AI in the web application that will look at policy and procedure documents and find policies and procedures that are relevant to the asset. This will help

users find what policies and procedures need to be followed for their asset. This will be developed using Python, spaCy[5] and Google Docs API[6].

1.4 Structure of This Document

The Structure of this document is as follows :

- Chapter 1 contains an introduction to this project, the motivation behind the project, the main feature and requirements of the project and the main contributions of this project.
- Chapter 2 contains research on the main topic of the project, a literature review and aims to describe what has and is being done.
- Chapter 3 outlines what the problem is that this project is trying to solve, what the objectives are, what are the functional requirements of the project and the non functional requirements.
- Chapter 4 describes how the project is going to be done, the architecture of the project, use cases, risk assessment, methodology, the plan for the implementation schedule, an evaluation on what was actually done and prototype of how the project is set to look.
- Chapter 7 contains the discussion, conclusion and future work of this project.

Chapter 2

Background

2.1 Thematic Area within Computer Science

The core topic of this project is Web Application for the management of Assets and their Associated Risks. There are many core areas that this project falls under, these areas are as follows Web applications, Security Management and Artificial Intelligence. The main area of computer science that this project falls under is Software Development. All of these will be discussed and expanded on in more detail in the rest of the chapter.

2.2 Project Specifics - Minimal Background Knowledge

2.2.1 Web Applications

Web applications is a software that is hosted on a web server often in the cloud. This means that whereas with other local applications do not need an internet connection, a web application needs an internet connection and is only accessible using a browser^[7].

2.2.2 Security Management

Security management is the identification of an organisation's assets such as buildings, people, machines, systems and information. This is then followed by the development, documentation and implementation of policies and procedures to protect the assets which were identified. Organisations use this information to assist in risk assessment and analysis, which helps identify threats and system vulnerabilities^[8].

2.2.3 Artificial Intelligence

Artificial Intelligence is an area of computer science which tries to develop computers which are capable of engaging in human like processes for example learning, reasoning and self correction[9].

2.2.4 Assets

An asset is anything that is of value to a company[10]. This can not only be something physical like a building or a computer but also data that the company stores and its cloud servers. The value to the company isn't always money, data for example if lost or damaged would not only cost the company money but also might damage their reputation.

2.3 Identify Community

The main areas that should be watched with this project are, Asset and Risk management, Security management and Artificial Intelligence.

Standards

One of the main things to be watched with asset and risk management is ISO standards. ISO standards are made by people who are experts in the field the standards are about[11]. ISO has many standards but for this project the main standard that this application would help implement is ISO 27001, ISO 27001:2013 is about protecting the security, confidentiality and availability of a company's information. The standard does this by identifying possible risks using risk assessment and then implementing methods to reduce those risks[12].

Books

Fundamentals of Enterprise Risk Management: How Top Companies Assess Risk, Manage Exposure, and Seize Opportunity[13].

By *John Hampton*

Fundamentals of Risk Management: Understanding, Evaluating and Implementing Effective Risk Management[14].

By *Paul Hopkin*

Conferences

RIMS 2021[\[15\]](#).

You'll find insight, opportunity and connections to build a bigger, stronger and more versatile risk management community.

Interop[\[16\]](#).

Interop provides a comprehensive and unbiased understanding of all the latest innovations, including cloud computing, virtualization, security, mobility and data centre advances that help position your company for growth.

Possible Interested Companies

More and more aspects of businesses are being moved online or digitised, like records being stored online instead of in a filing cabinet, customer data needing to be stored online or businesses now needing websites to sell items and seem more legitimate. Though this in some ways makes record keeping and storing data easier for companies, it also opens them up to new risks and threats they didn't previously have to deal with. Hearing about data breaches used to be something that only happened now and again but now it has become all too familiar to hear about data breaches or malware affecting all sorts of companies[\[10\]](#).

And no longer is it just tech companies being affected by these data breaches or malware attacks, these attackers are now attacking not only large tech companies but also individuals, smaller businesses and government run businesses. Both, very large companies and even the smallest of startups or small shops are now being affected and these companies and businesses now need to think more seriously about their assets, risks and security. The repercussions of these attacks can be dire as evident by the attack on Ireland's health care system (HSE) in 2021, a proper assessment of its assets and the risk associated with that asset could have possibly helped limit the scope of this attack[\[17\]](#).

For the above mentioned reasons this project could be useful for not only larger companies like the HSE or big tech companies like Google or Apple but also for small startups and regular small businesses that are self-owned that maybe don't think much about security, assets and their risks but in the wake of all these attacks now need to start considering it. Kaspersky Lab by 2018 found that 30.01 % of computers on the internet at least once were attacked by Malware class attack; ransomware attacks are reflected on computers of 765538 unique users[\[18\]](#); these statistics highlight the need for even small businesses to start thinking more seriously about their assets.

2.4 A Review of Management of Assets and their Associated Risks

In this section other work that has been done will be discussed in detail. Subsection 1 will talk about relevant papers in detail and compared them to this project. Then in subsection 2 other similar applications features will be discussed and compared to this project.

2.4.1 A Literature Review of Management of Assets and their Associated Risks

In Information Security Risk Management[10] the authors discuss the application of information security risk management methods in enterprises. The authors do this by looking at examples of information security risk management and making recommendations on how businesses can better protect themselves.

The paper defines Information as “an important asset that is essential to the organization’s business needs”, and defines information security as “preservation of confidentiality, integrity and availability of information”. Information security is based on 3 categories described in ISO 27001:

- Confidentiality - property that information is not made available or disclosed to unauthorized individuals, entities, or processes.
- Availability - the property of being accessible and usable upon demand by an authorised entity.
- Integrity - property of accuracy and completeness.

Businesses these days need to invest in the development of an information security management system, they can do this by identifying and assessing the risks and then developing a plan to protect themselves against those risks.

The authors describe the Swiss Cheese Model method which allows the businesses to identify assets threats and develop an action plan. When using this method the business needs to compile a register of assets containing the assets location and owner, determine what is being protected and then when that is done analyse the threats that could end up damaging the asset. The solution should not only be a technical solution but also use organisational methods. The model then evaluates vulnerabilities for each protection level.

Though larger companies have more to lose, smaller businesses still need to protect themselves and they can look at larger companies as an example and make changes to suit their size. The paper gives 3 examples of this, they are detailed in the table below.

Threat	Damage	Control Measure
Hacking a financial account	Theft of cash	The separation of business and personal
An external cyber attack on company data.	Loss of integrity and confidentiality of a part of data.	An integrated approach to data protection of all kinds.
Data leakage from a company employee	Loss of privacy of a part of data	Access to information assets based on official need.

TABLE 2.1: Small Business Threat Examples

The paper concludes that smaller businesses need to learn to be making the right decisions when it comes to information security management, they can learn this by looking at larger companies. Though the cost is not as big as it may seem there is still a cost and the authors suggest thinking of this cost like you would insurance where it can end up saving the business money[10].

In Risk Assessment on Information Asset an academic Application Using ISO 27001[19] the authors aim to determine the risk in information assets and business impact to a university[19]. There are 3 stages to the research, first categorising information assets that might pose a threat to information security, second calculating the asset value and finally carrying out a risk assessment in relation to the threat.

The paper uses qualitative data analysis, using part of element ISO 27001 and analysis of the data. The data for the study was collected using a questionnaire, the two types of data used were primary which consisted of documentation and in depth interviews, secondary data consistent of journals, papers, websites etc.

The research used a questionnaire which was adapted from ISO 27001 asset management on clausal access control to identify asset value and risk assessment criteria. The first stage of research was spent finding issues about information security through academic application this is done through observation, reading literature and interviews. The next step was collecting data through interviews and questionnaires. After the data is collected asset analysis and business impact analysis is made based on the questionnaire.

Once the assets have been identified they are then classified by the kind of asset and hardware and then the identification of the risks. When this is completed the next stage is to calculate the value of the assets based on the three aspects of information security, confidentiality, integrity and availability.

The formula is as follows:

$$\text{ValueAsset}(AV) = CV + IV + VV \quad (2.1)$$

Where CV is Confidentiality Value, IV is Integrity and AV is availability.

The next stage in the research was to identify the threats or weaknesses of the assets and then calculate the chance of these threats and weaknesses happening. This was done based on IT managements guide of 3 probabilities Low (0.0 - 0.3), Medium (0.4 - 0.6) and High (0.7 - 1.0). This is then followed by carrying out a Business Impact Analysis, the result of which determines the tolerance of the assets. After this the risk level is identified by looking at the probability threat level and the level of impact of the risk.

The paper concluded that risk management is a substantial problem the organisation and the responsibility of IT governance and computer security. They found that academic application had a high risk but user data and passwords were of the highest value and have a critical risk. They advised the university should have managed and mitigated the risks on the application. The research does mention that due to a lack of literature it could only use three variables and future work should contain more variables for a new model for risk assessment on information security[19].

In Developing a Whole Company Culture of Asset Management[20] the authors describe a water company's journey and the challenges they faced when the company decided to develop its asset management.

The authors begin this paper by discussing how the company in 2006 started to review it's asset management capabilities so they could identify where the company could use development in asset management. The review led to the following initiatives:

- The first was changing the organisational structure of the company so that asset management roles and responsibilities were more clearly defined.
- The second was developing an asset management framework which the company would use to make policies, plans, strategies and provide a line of sight throughout the whole company so that every level of the company could see what was being done with assets.

- The final initiative was to implement a risk-based approach for asset intervention to make sure that the investment budget was being spent correctly and that this could be demonstrated.

The paper then discusses each of these initiatives in more detail. The company's original organisational structure which was focused on maintenance and repairs was switched to one that was more focused on the optimization of the management of assets. This new structure provided a deeper understanding of the assets requirements and the ability to find a balance between risk, cost and performance of the assets. The new structure defined 3 asset management roles and responsibilities:

- Asset Owner who is responsible for investment, strategic asset planning corporate risk. They are also responsible for setting and maintaining standards of services, budget and regulatory compliance.
- Asset Operator who is responsible for the day to day maintenance and operation of the assets.
- Asset Deliverer who is responsible for the design and construction of new assets as well as major refurbishments of already existing assets.

The asset management framework was then discussed in more detail by the authors. Previously the company had been managed by a separate ISO 9001 departmental management system. The new asset management framework provided a clear line of sight along with coherence between the companies vision, operational maintenance and investment and asset management strategies and plans align with controls to ensure environmental and customer standards.

The paper next discusses the new risk-based prioritisation of maintenance and investment to asset management that the company took. The aim of this new approach was to improve not only the management of risks but also the understanding of risks. An iterative approach was taken to build on external initiatives from environmental regulators.

Modelling and analytic tools were developed so that operational maintenance response and investment could be prioritised by allowing a consistent way of assessing the impact and likelihood of a service failure across all the company's assets at different levels, along with a review of all emerging risks from different areas including environmental, customer and financial. This also allowed the reporting of high-level risks and mitigation measures to senior management.

The heat map report provided a representation of the severity of the risk to the company's service standards and industry drivers. The position on the map was based upon a combination of financial, legal, environmental, customer, regulatory and health and safety impacts along with probability assessments, performance, condition, potential hazards and predictive modelling. The asset risks at operational and tactical levels are combined with other risks levels to understand the full business risk.

The paper concluded that improved asset management capabilities have led to significant benefits for the company. The focus on organisation structure, asset management frameworks and consistency in risk management principles have been critical to the success of this strategy. The improved decisions on asset intervention have increased certainty about not only the resilience of current service levels but also the company's ability to deliver expected services[20].

In Current challenges in information security risk management[21], the authors give an overview of current risk management approaches, compare these approaches and then discuss the following questions: What are the common problems at implementing current information security risk management approaches and to what extent are these problems addressed by research? How can information security solutions costs be factored in when determining the risk mitigation strategies, i.e making the appropriate risk versus cost trade-off?

The authors start by giving an overview of the different security risk management approaches. There are 8 approaches in total that the authors give an overview of, NIST SP 800-30, ISO 27005, EBIOS, OCTAVE, CRAMM, FAIR, ISAMM, ISF.

The authors then discuss that all the methodologies mentioned are very similar and all require inventory and security classification of relevant infrastructure elements and identification of the organisations mission and goals. In the first phase and all the following phases seem to require the identification of threats and vulnerabilities to determine the threat probability along with already implemented controls. All of the methodologies sharing a lot of commonalities allowed the authors to create a generic information security risk management methodology. This generic methodology consists of the following:

- System Characterisation: Definition of system boundaries and assets used and or required by the system requires a systematic inventory of all assets and the determination of an acceptable risk level for each asset.

- Threat and Vulnerability Assessment: Determine the potential threats, corresponding threat origins and vulnerabilities. This phase requires a security requirement checklist which can be used for compliance evaluation regarding current and planned controls.
- Risk Determination: Determine the probability of a threat exploiting a certain vulnerability in the system. An impact analysis then determines the impact on the organisation's ability to perform its mission, if a threat should successfully exploit a certain vulnerability.
- Control Identification: This phase considers already implemented controls, additional control implementations which could mitigate or reduce risks to an acceptable level.
- Control Evaluation and Implementation: Evaluate identified controls implementations or combinations regarding their cost to benefit ratio. The controls that are suitable to mitigate the risk and have the lowest cost are incorporated into the control implementation plan.

This allowed the authors to align the problem and solution identification to its generic phases and ensure the research results could be applied to a broad range of existing information security risk management methodologies.

The authors next go on to discuss the current challenges and potential solutions in security risk management. The risks are based on the risk management and decision support approaches described already in the paper, related work and feedback from the scientific and professional community.

Challenge 1 is asset and countermeasure inventory, which is the problem concerning proper identification and inventory of information technology assets. Everything connected in any kind of technology component has to be seen as an asset and the dynamic and changes in the IT environments increase the complexity of this problem. The authors then discuss a possible solution to this challenge based on Moneteruno and Fenz[22] and Fenz et al.[23].

Challenge 2 is assigning asset values, which is the problem with assessing the value of the assets especially when assessing small items. Assessing other values such as the loss due to system downtimes are also difficult to assess. The authors then discuss a possible solution to this challenge based on Fenz et al.[24].

Challenge 3 is the failed prediction of risk, this challenge is based on the ever-changing nature of risks making it virtually impossible to know which asset will be of interest to an attacker.

Challenge 4 is the overconfidence effect which is that managers tend to assume risk estimations far too optimistically. Combined with the stress decision-makers face and the limited time and resources this can lead to a dismissive attitude towards formalisms. Rhee et al[25].

Challenge 5 is knowledge sharing, which is that knowledge sharing between experts would be desirable in order to prevent developing the same security programs by independent people and organisations over and over again. This sharing of knowledge could also lead to higher quality security programs.

Challenge 6 is risk vs cost trade-offs, where the development of countermeasures is usually motivated by risk management and technical effectiveness is enforced to minimise perceived risks for the assets. The cost-effectiveness of the countermeasure is often neglected and should not exceed the expected loss of an asset. The authors then discuss numerous approaches to this challenge.

The authors then discuss how the methodologies mentioned before address these challenges. They mention that most of the methodologies cover challenges 1 to 3, challenge 4 is unable to be solved with a technical solution the same is said of challenge 5. Challenge 6 can not be solved by most of the mentioned methods due to their generic nature and orientation on national and international standards.

The authors then conclude that common challenges when implementing information security risk management approaches were identified in the areas of asset and countermeasure inventory identification, asset value assignment, risk prediction, overconfidence effect, knowledge sharing and risk vs cost trade-off. They were not able to find solutions that fully cover the problems identified to a satisfying level[21].

2.4.2 Practical Applications Review of Management of Assets and their Associated Risks

ISMS.online

ISMS.online is a cloud-based management system for information security, privacy, business continuity and compliance work. ISMS.online helps companies achieve globally recognised certifications like ISO 27001 and assists them comply with data protection regulations like GDPR [26]. ISMS online is a paid application, it helps you improve your Information Security Management System, Privacy Information Management System, Business Continuity Management System and Integrated Management System. Some of the features it offers are as follows[26] :

- Policies & Controls Management
- Risk Management
- Measurement & Automated Reporting
- Audits, Actions and Reviews
- Mapping and Linking work
- Interested Party Management
- Documented Procedures
- Other Standards and Regulations
- Staff Awareness and Compliance Assurance
- Supply Chain Management
- User Management and Permissions
- Privacy and Security

CyberComply

CyberComply is a tool provided by Vigilant Software, that allows its users to manage their cyber security and data privacy obligations^[27]. CyberComply packages together multiple Vigilant Software products, vsRisk, Compliance Manager, Data Flow Mapping Tool, DPIA Tool and GDPR Manager. CyberComply is a subscription-based application available for a single user or multiple users, through monthly and annual subscriptions. The tool is hosted online and accessible through a compatible browser. Some of the features offered by CyberComply are as follows^[27]:

- Quickly identify and treat data security risks
- Map data flows
- Manage DSARs, DIPAs and data breaches lawfully
- Keep track of data security compliance requirements
- Reduce errors and improve the completeness of risk management processes
- Follow step-by-step processes to ensure compliance
- Intuitive and interactive dashboard

2.4.3 Review Summary

Though the papers found for this paper were not about using a web application to manage company assets and associated risks. This project could help implementing and maintaining the risk and asset management methods described in the papers. The papers also gave more insight into what the application could do in order to help companies better manage their assets and associated risk.

The advantage this project has over the applications and tools described in the practical application section above are that both of the products are paid. There is no information on how much ISMS.online costs a request for a quote needs to be made[26]. CyberComply is subscription-based and depending on the number of users the price can vary from £199.95 for one user to £1,199.95 for 15 users per month or an annual subscription of £1199.95 for one user and £11,999.95 for 15 users[27].

The price of these applications makes them less attainable for smaller businesses or startups who may need to prioritise putting money into other areas of the business. This project aims to be open source so that smaller businesses and startups don't need to worry about the cost of the application, the only cost these businesses will face would be hosting it through Firebase which most smaller businesses could use the free version. This project will also offer templates allowing users to have an easier time getting set up on the system. Another advantage of this project over ISMS.online and CyberComply is the AI, this AI will make suggestions to users about policies and procedures their company has that may apply to the asset.

Chapter 3

Problem - Web based application to manage company assets and their associated risks.

3.1 Problem Definition

This project aims to create a web application that will help companies to manage their assets and the risks associated with each asset. For this application an asset is anything that is of value to the company, this could be a server, a building or data.

There are many regulations that companies need to follow depending on the type of asset, for buildings there can be fire regulations and safety regulations. For assets like user data companies have to make sure they follow regulations like GDPR. Companies need to make sure they are following these regulations which can sometimes be a challenge. The application developed in this project aims to help companies with this by allowing them to see the regulations that apply to an asset, view when the regulations were last reviewed by an employee and also more easily check they are following regulations.

Smaller businesses and startups aren't exempt from these regulations and when starting a business, trying to figure out how to make sure you are following them can be a difficult task. For some businesses who have little to no experience with technology, the regulations around data and technology can be quite daunting especially since some of them are so recent.

The web application hopes to help small businesses and startups with this issue by having templates for users to use. There will be templates for different types of assets as well as basic company startup templates for assets most companies will have. The template will

have suggested protections against risks and policies to comply with regulations along with suggesting regulations that may apply to the asset. These templates will also help bigger companies with more experience get set up and protected faster.

Another feature of the application that will help companies keep on top of their assets, is an AI. This AI will look at policy and procedure documents uploaded by the company and find policies and procedures that should be applied to the asset. This will help employees of the company protect assets.

What will set this project apart from other similar projects will be the cost. Asset management software can be very expensive and for smaller companies or startups this cost may be too much. No matter the size of the company, asset management is important but can be expensive, this project if it is made available to the public will be free for companies to use.

3.2 Objectives

The objectives this project hopes to achieve are as follows:

- The database is well designed and efficient
 - To have an application that is quick and easy to use, a well-designed database is needed so that data can be displayed to the user efficiently.
- Web Application is user friendly
 - As this application will be used by people with varying degrees of knowledge with technology, the application needs to be easy to use. The data on the assets, risks and controls should be displayed clearly and the application should be easy to navigate.
- Web application is hosted and available through any browser
 - The application will need to be hosted so that users can access the application through their chosen browser.
- Users can log in and sign up
 - Users will be able to securely log in and sign up to the application so that they can view the company's assets, risks and controls.

- Users can view all their assets, associated risks and controls
 - Once logged in a user can view all the assets, risks and controls they have permission to see.
- Users will be able to edit and manage any asset that they have admin access to.
 - Users should only be able to edit and delete the assets they have ownership over.
- Users should be able to manage controls and risks.
 - If the user has the correct permissions, the user should be able to edit or delete any control or risk in the application.
- Companies will be able to upload policy documents that the AI can use to make suggestions.
 - Companies should be able to upload documents to the application so that the AI can scan the documents and make suggestions to the user.
- AI that can make suggestions about policies to the user.
 - The AI should look at documents uploaded by the company and make suggestions to the user on which policies of the company may apply to the asset.
- Templates for initial set-up will be available to all users

3.3 Functional Requirements

The functional requirements of this project are as follows:

- Application allows users to login securely
- Application allows users to logout securely
- View assets and their details
- View details of risk
- View control details
- View list of controls
- View list of assets
- View and use templates

- Add new assets, risks and controls
- Delete assets, risks and controls
- Update and review assets, risks and controls
- User can only view their companies data
- Admin login available
- Admin can add new organisations
- Company Admin can give the AI access to policy and procedure documents
- AI will suggest policies and procedures for assets
- Website will be hosted and available through any browser
- Data will be stored in a cloud database

3.4 Non-Functional Requirements

The non-functional requirements of this project are as follows:

- Data is encrypted and only accessible to administrators
- Web application will fit multiple screen sizes
- Web application is user friendly
- Web application is reliable
- The database will be backed up daily or hourly
- Web application is accessible
- Whole system needs to be scalable

Chapter 4

Implementation Approach

4.1 Architecture

In this section an overview of the project architecture will be given, along with the technologies used and a high level view of the database.

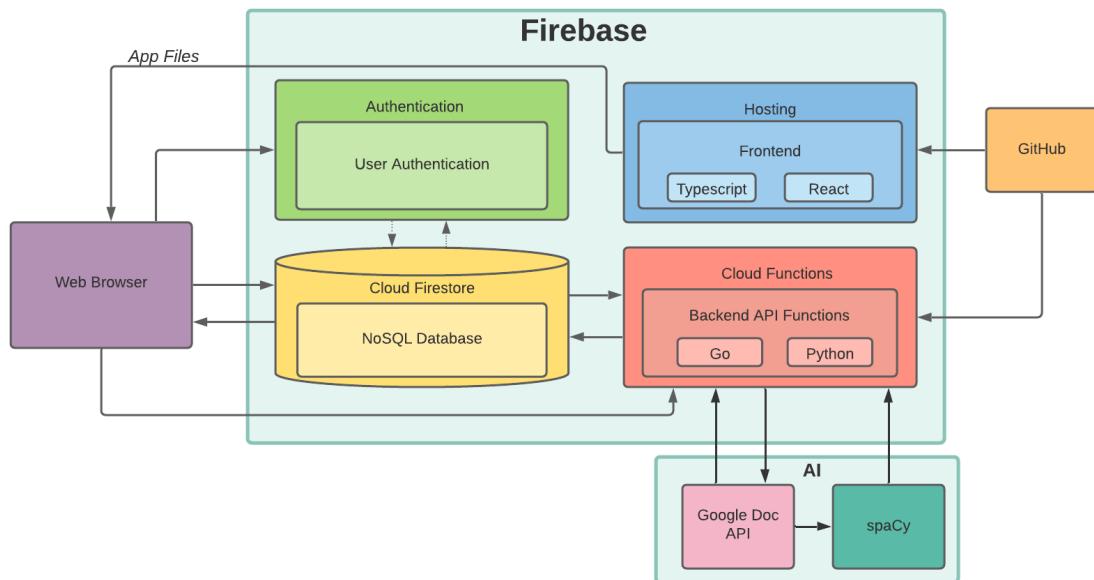


FIGURE 4.1: High Level Architecture Diagram for Web Application

4.1.1 Technologies Involved

- Firebase Hosting
 - Firebase Hosting will be used to make the application available online for users to access through their browser. Firebase Hosting is a product that provides production grade web content hosting for developers[28].
- Cloud Firestore
 - Cloud Firestore will be used to store the data for this project. Cloud Firestore is a flexible and scalable NoSQL cloud database[29].
- Cloud Functions
 - Cloud Functions will be used for the backend of the application. Cloud Functions is a serverless framework that allows you to run backend code in response to events triggered by HTTPS or other Firebase features[30].
- Firebase Authentication
 - Firebase Authentication will be used to handle user authentication need for the web application. Firebase Authentication provides SDKs and backend services to authenticate users in app[31].
- Git Hub
 - Git Hub will be used for version control for this project. Git Hub is a code hosting platform for version control using Git[32].
- React
 - React will be used to build the frontend of the application. React is a JavaScript library used for building user interfaces[2].
- Type Script
 - Type Script is the programming language that will be used to write the frontend of the application. Typescript is a strongly typed programming language that builds on JavaScript[1].
- MUI Core
 - MUI Core will be used to help build the application. MUI Core provides a library of components in order to help you build React applications[3].

- Visual Studio Code
 - Visual Studio Code will be the IDE used to develop the application. Visual Studio Code is a source code editor which runs on desktop[33].
- Go
 - Go is the programming language that will be used to write the backend. Go is an open source programming language supported by Google[34].
- spaCy
 - spaCy will be used to develop the AI within the application. spaCy is a free, open-source library for advanced Natural Language Processing in Python[5].
- Python
 - Python will be used together with spaCy to develop the AI of the web application. Python is a high level programming language.
- Google Docs API
 - Google Docs API will be used to read documents in plain text so that the information can be used by spaCy[6].

4.1.2 File Structure

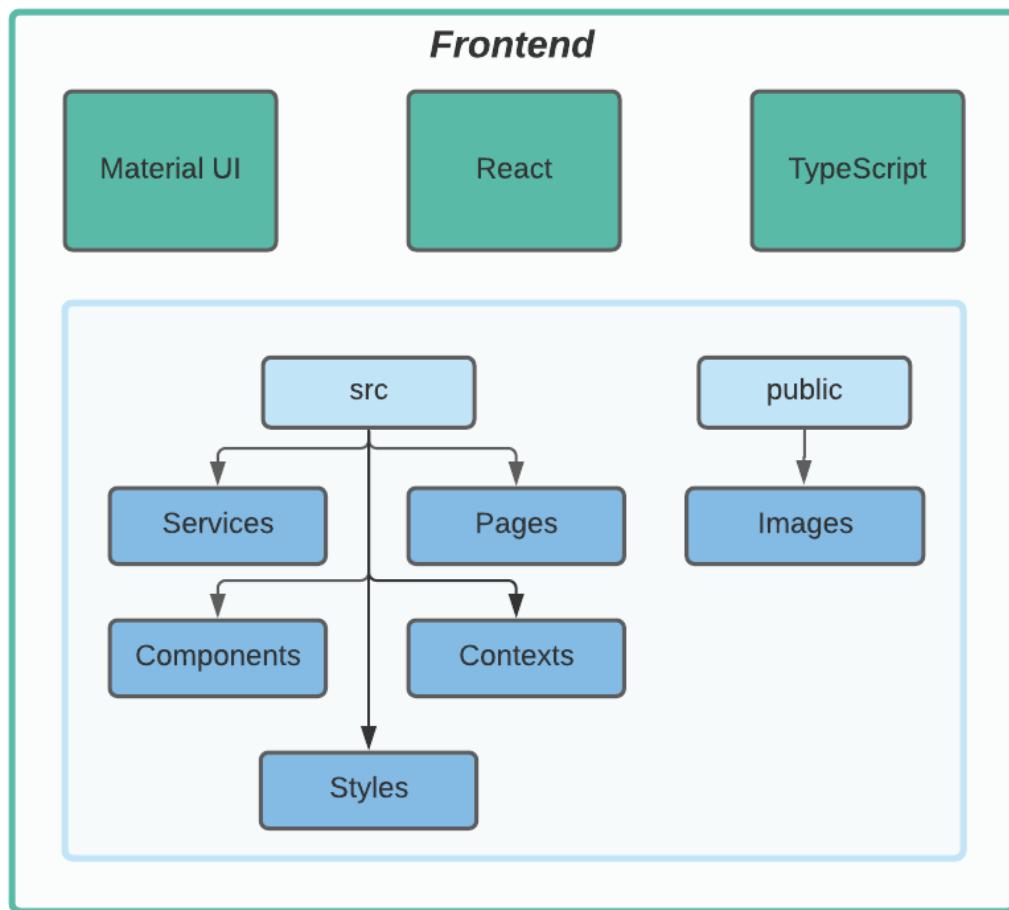


FIGURE 4.2: High Level Overview of the File Structure for Web Application

The frontend of this project will be developed using React, Typescript and MUI. There is no standard file structure used for react and therefore the file structure is left up to the developer. The file structure for this project was chosen because of the authors previous experience with this structure in React projects. The project will have two main folders src and public, the public folder will contain an images folder which has all the images for the web application. The src folder will have multiple sub folders, these folders are as follows.

- The Services folder will contain all the code required to connect to the Cloud Firestore database and Cloud Functions.
- The Pages folder will contain all the pages of the application.

- The Components folder will have all the components which will be used throughout the project.
- The Context folder will contain code that multiple components will need like the api or system messages.
- The Styles folder will have all the CSS for the components and pages.

4.1.3 Database Overview

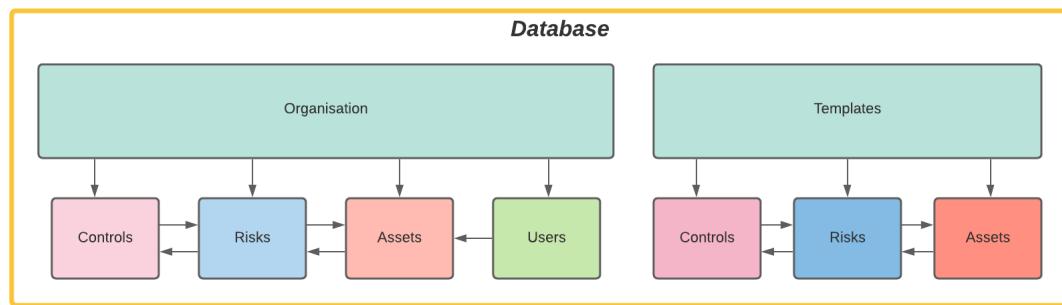


FIGURE 4.3: High Level Overview of the Database for Web Application

The database used for this project will be Cloud Firestore it is a NoSQL database. Cloud Firestore was chosen as it is seamlessly integrates with other Firebase products which are being used in the project. Cloud Firestores NoSQL database uses collections of documents to store data, there will be two documents in this project[29]. One document will be Organisations which will store data on the users belonging to that organisation and the organisations assets, risks and controls. The second document is Templates, this document will store the assets, risks and control information for templates.

4.2 Use-Case Description

This section contains use case descriptions for the web application.

1. Login	
Summary	User logs into the application
Pre Conditions	User must have a working internet connection and have a browser open
Primary Actors	Employee of Registered Organisation
Secondary Actors	System
Flow of Events	<p>Employee: Enters the website URL. System: Loads the login page. Employee: Enters valid email and password. Employee: Clicks Login button. System: Checks with Firebase Authentication and Cloud Firestore that the Employee details are correct System: Brings Employee to home page.</p>
Alternative Flow	<p>Employee: Enters the website URL. System: Loads the login page. Employee: Enters email and password. Employee: Clicks Login button. System: Checks with Firebase Authentication and Cloud Firestore that the Employee details are correct. System: Password is incorrect. System: Displays error message. Employee: Enter valid email and password. Employee: Clicks Login button. System: Checks with Firebase Authentication and Cloud Firestore that the Employee details are correct System: Brings Employee to home page.</p>
Post Conditions	Employee is now on home page

TABLE 4.1: Use Case 1 - Login

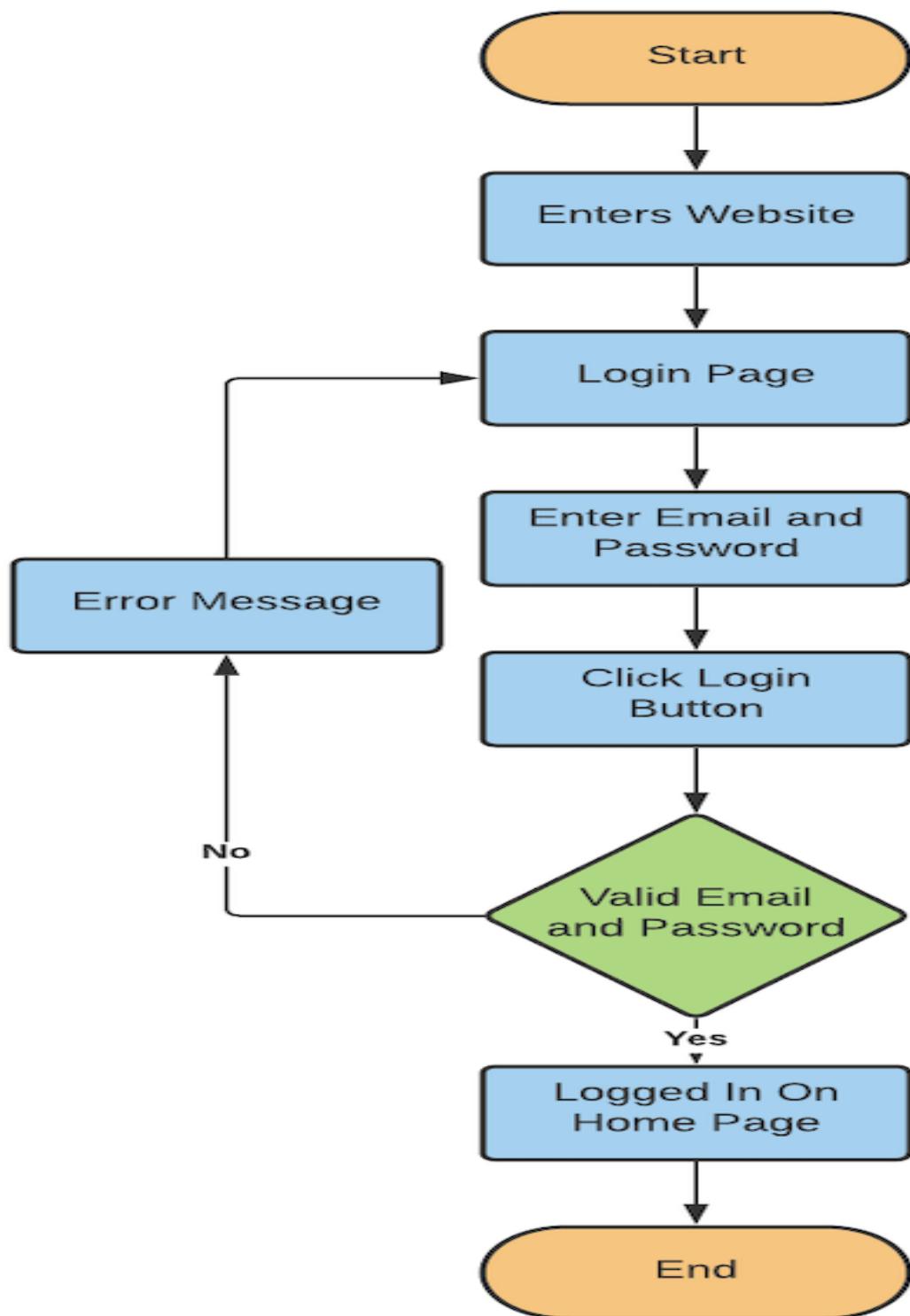


FIGURE 4.4: Flow diagram for use case 1

2. Maintain Asset	
Summary	Employee adds, updates and deletes asset.
Pre Conditions	Employee is logged in and has permission to update, delete and add assets, Employee is on the home page.
Primary Actors	Employee of Registered Organisation
Secondary Actors	System
Flow of Events	<p style="text-align: center;">Add</p> <p>Employee: Clicks on Assets in the navbar. System: Brings the employee to the assets list page. Employee: Employee clicks add asset button. System: Brings employee to the add asset page. Employee: Fills in details of the asset. Employee: Clicks add asset button. System: Adds new asset to database. System: Displays success message to employee. System: Brings employee to the assets list page.</p> <p style="text-align: center;">Update</p> <p>Employee: Selects Asset they want to edit from the table. System: Brings employee to the asset page Employee: Clicks edit button. System: Opens edit asset pop up. Employee: Changes the details of the asset. Employee: Clicks confirm button. System: Connects to Cloud Firestore and makes changes. System: Updates detail on assets page System: Closes pop up and displays success message.</p> <p style="text-align: center;">Delete</p> <p>Employee: Selects Asset they want to delete from the table. System: Brings employee to the asset page. Employee: Clicks delete button. System: Displays warning message, asking employee if they are sure. Employee: Clicks confirm button. System: Connects to Cloud Firestore and deletes asset. System: Displays success message and brings employee back to asset list page.</p>
Alternative Flow	<p>System: Fails to update, delete or add asset System: Error message shown to employee</p> <p style="text-align: center;">OR</p> <p>Employee: Cancels Add, Edit or Delete action. System: Closes pop-ups and brings employee to previous page.</p>
Post Conditions	Asset is added, Asset is updated, Asset is deleted

TABLE 4.2: Use Case 2 - Add, Edit and Delete Asset

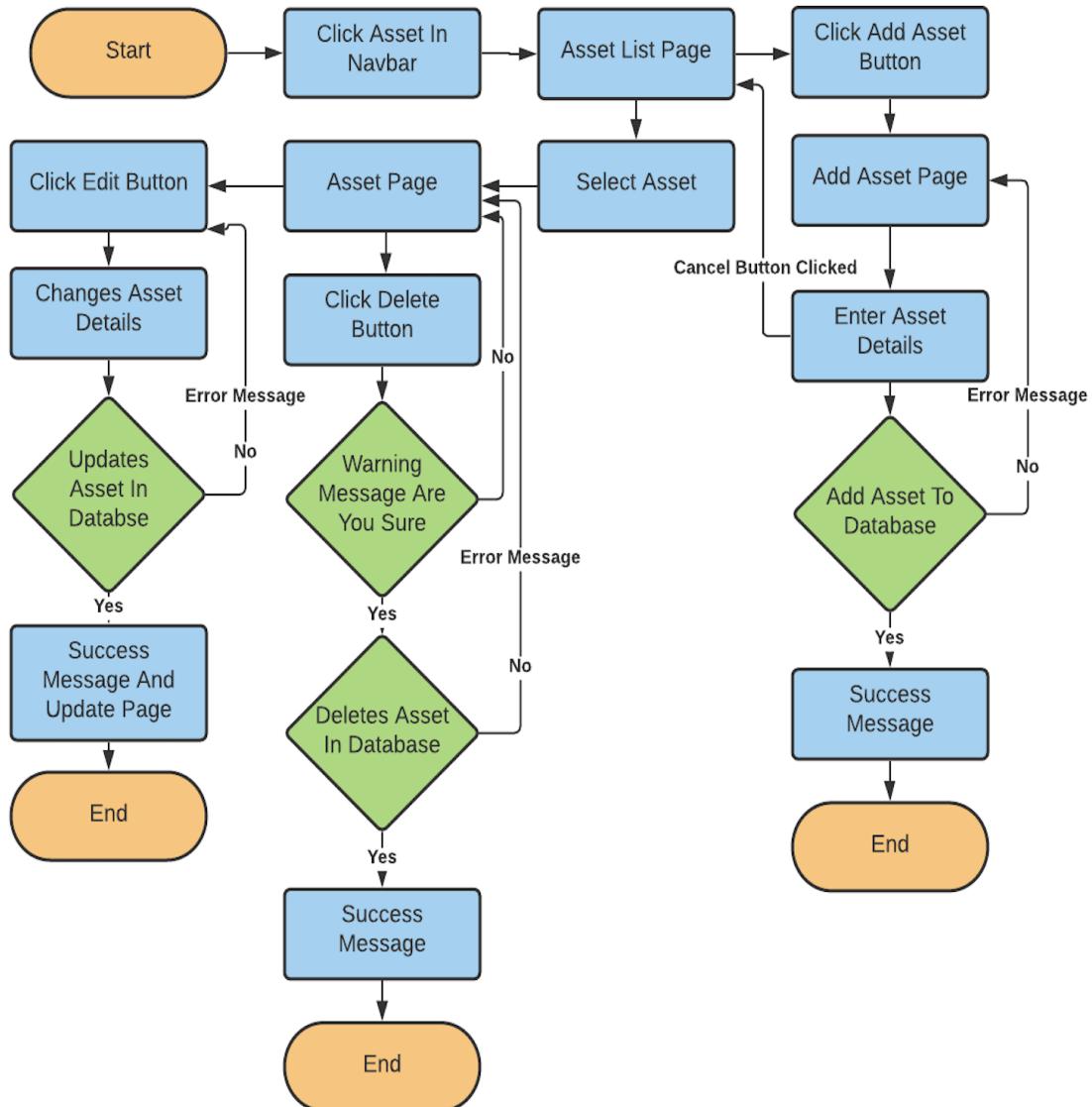


FIGURE 4.5: Flow diagram for use case 2

3. Maintain Risk	
Summary	Employee adds, updates and deletes a risk.
Pre Conditions	Employee is logged in and has permission to update, delete and add risks, Employee is on the home page.
Primary Actors	Employee of Registered Organisation
Secondary Actors	System
Flow of Events	<p style="text-align: center;">Add</p> <p>Employee: Clicks on Assets in the navbar. System: Brings the employee to the assets list page. Employee: Clicks the asset they want to add the new risk to from the table. System: Brings employee to the asset page. Employee: Clicks on the add risk button. System: Displays the create risk pop up. Employee: Enters the details of the risk and clicks confirm button. System: Connects to Cloud Firestore and makes changes. System: Displays success message to employee. System: Brings employee to the assets page.</p> <p style="text-align: center;">Update</p> <p>Employee: Selects Asset they want to edit a risk for from the table. System: Brings employee to the asset page Employee: Clicks the risk they wish to edit from the list of risks on the asset page. System: Brings the employee to the risk page. Employee: Clicks edit button. System: Opens the edit risk pop up. Employee: Changes the details of the risk, clicks confirm button. System: Connects to Cloud Firestore and makes changes. System: Updates detail on page and displays success message.</p> <p style="text-align: center;">Delete</p> <p>Employee: Selects Risk they want to delete from the table. System: Brings employee to the risk page. Employee: Clicks delete button. System: Displays warning message, asking employee if they are sure. Employee: Clicks confirm button. System: Connects to Cloud Firestore and deletes risk. System: Displays success message and brings employee back to asset page.</p>
Alternative Flow	<p>System: Fails to update, delete or add asset System: Error message shown to employee</p> <p style="text-align: center;">OR</p> <p>Employee: Cancels Add, Edit or Delete action. System: Closes pop-ups and brings employee to previous page.</p>
Post Conditions	Risk is added, Risk is updated, Risk is deleted.

TABLE 4.3: Use Case 3 - Add, Edit and Delete Risk

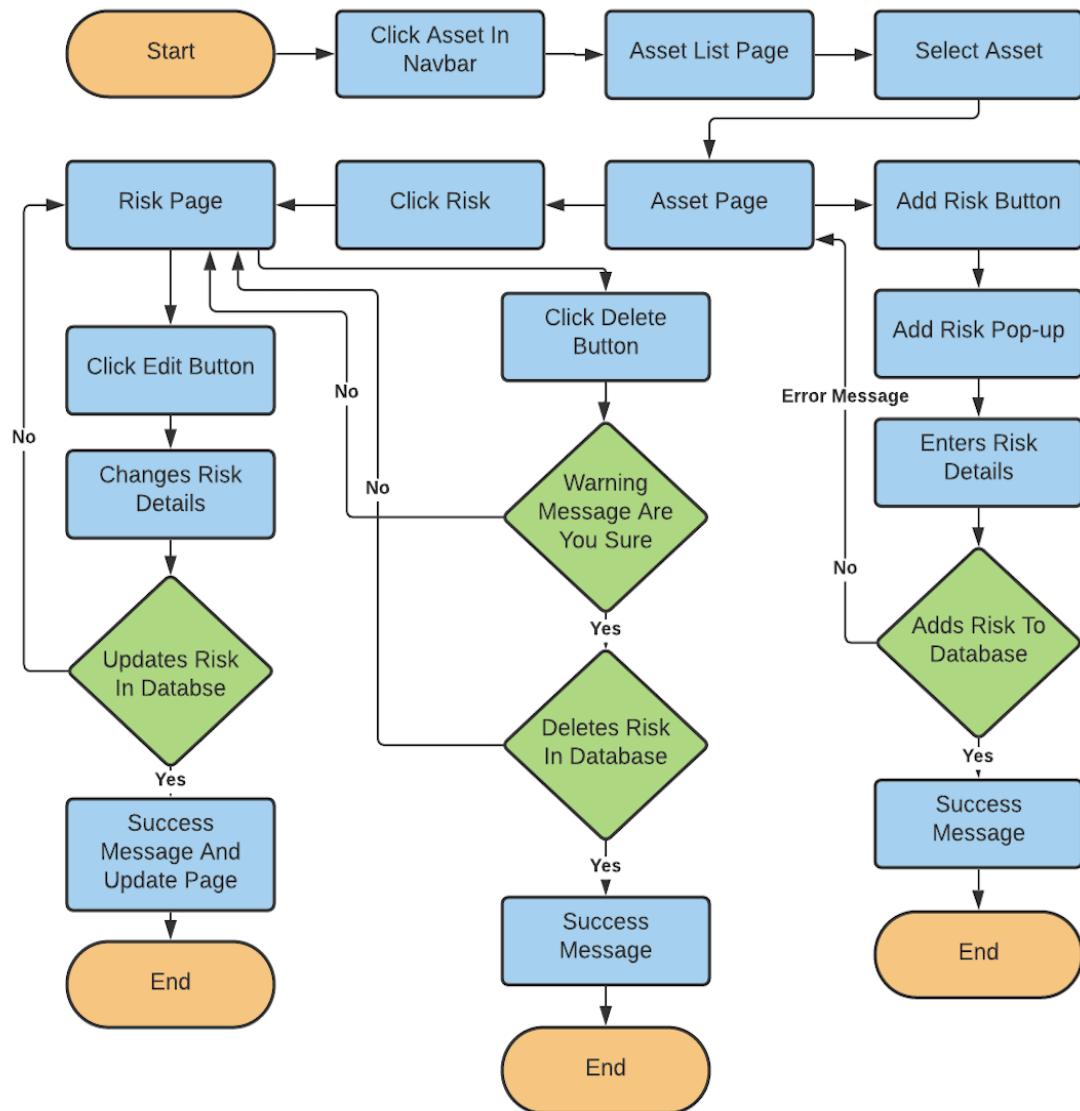


FIGURE 4.6: Flow diagram for use case 3

4. Maintain Control	
Summary	Employee adds, updates and deletes a control.
Pre Conditions	Employee is logged in and has permission to update, delete and add controls, Employee is on the home page.
Primary Actors	Employee of Registered Organisation
Secondary Actors	System
Flow of Events	<p style="text-align: center;">Add</p> <p>Employee: Clicks on Controls in the navbar. System: Brings the employee to the controls list page. Employee: Employee clicks add control button. System: Brings employee to the add control page. Employee: Fills in details of the control. Employee: Clicks add control button. System: Adds new control to database. System: Displays success message to employee. System: Brings employee to the control list page.</p> <p style="text-align: center;">Update</p> <p>Employee: Selects Control they want to edit from the table. System: Brings employee to the control page Employee: Clicks edit button. System: Opens edit control pop up. Employee: Changes the details of the control. Employee: Clicks confirm button. System: Connects to Cloud Firestore and makes changes. System: Updates detail on page and displays success message.</p> <p style="text-align: center;">Delete</p> <p>Employee: Selects control they want to delete from the table. System: Brings employee to the control page. Employee: Clicks control button. System: Displays warning message, asking employee if they are sure. Employee: Clicks confirm button. System: Connects to Cloud Firestore and deletes control. System: Displays success message and brings employee back to control list page.</p>
Alternative Flow	<p>System: Fails to update, delete or add control System: Error message shown to employee OR</p> <p>Employee: Cancels Add, Edit or Delete action. System: Closes pop-ups and brings employee to previous page.</p>
Post Conditions	Control is added, Control is updated, Control is deleted.

TABLE 4.4: Use Case 4 - Add, Edit and Delete Control

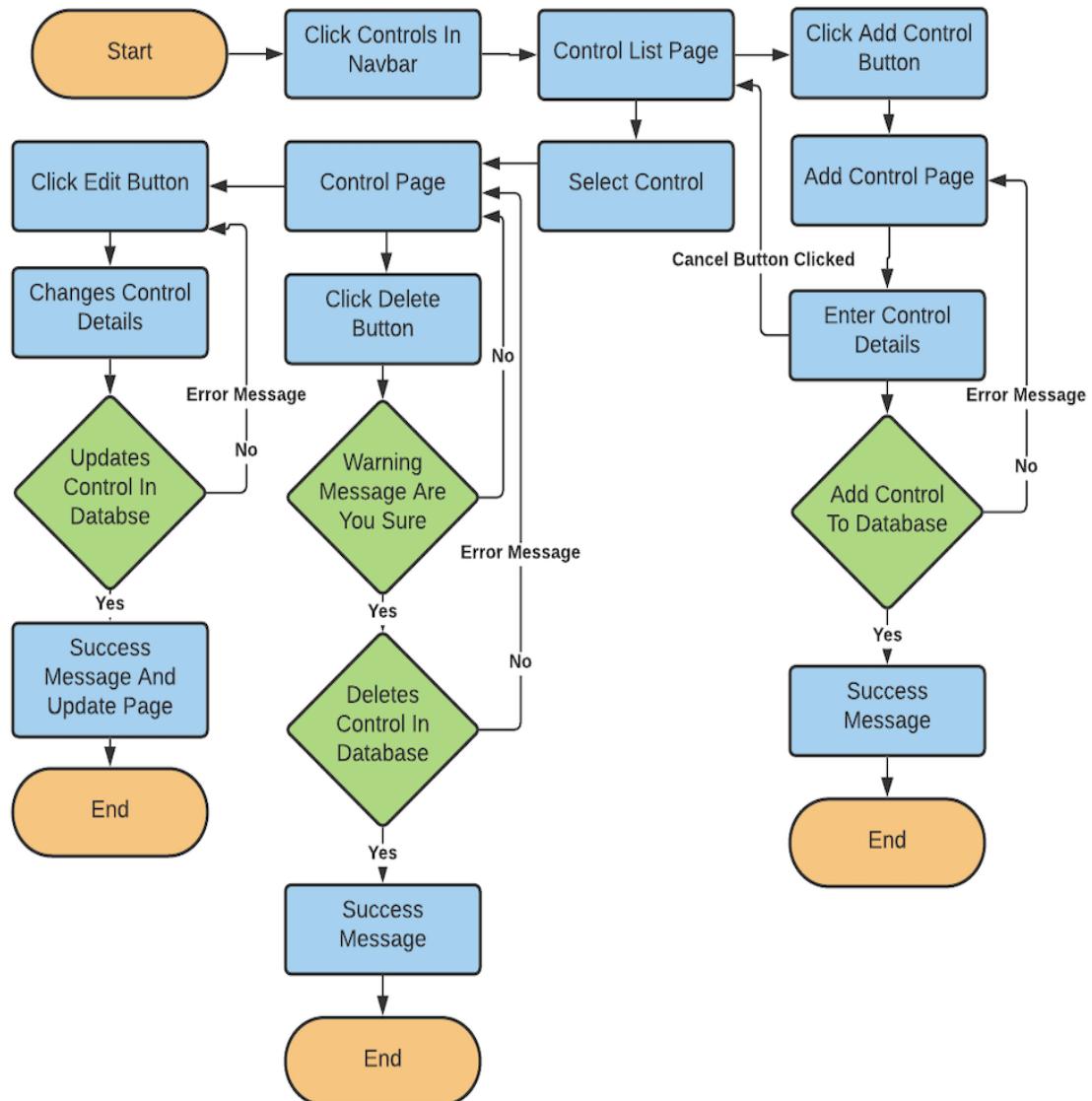


FIGURE 4.7: Flow diagram for use case 4

5. View Assets, Risks & Controls	
Summary	Employee views asset, risks and controls
Pre Conditions	Employee is logged in and is on the home page.
Primary Actors	Employee of Registered Organisation
Secondary Actors	System
Flow of Events	<p>Employee: Clicks on Assets in the navbar. System: Brings the employee to the assets list page.</p> <p>Employee: Clicks on the asset they would like to view System: Brings employee to asset page</p> <p>Employee: Clicks risk in the list of associated risks they would like to view in more detail. System: Brings the employee to the risk page.</p> <p>Employee: Clicks control in the list of associated controls they would like to view in more detail. System: Brings the employee to the control page.</p>
Alternative Flow	<p>Employee: Chooses control in the navbar System: Brings employee to control page</p> <p>Employee: Clicks the control they would like to view System: Brings the employee to the control page.</p> <p>Employee: Clicks the associated assets or risks they would like to view in more detail System: Brings the employee to the asset or risk page.</p>
Post Conditions	Employee has been able to view assets, risks and controls

TABLE 4.5: Use Case 5 - View Assets, Risks and Controls Control

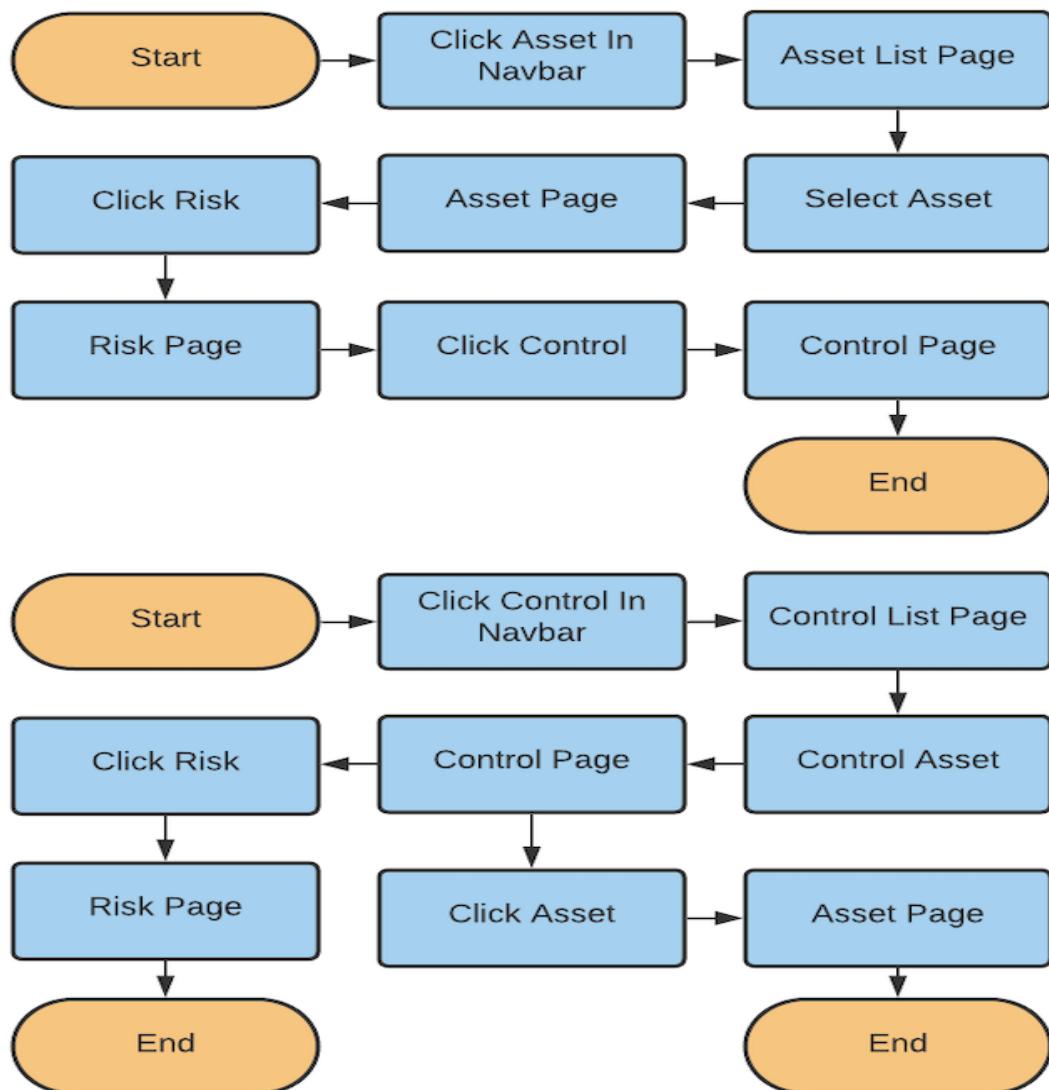


FIGURE 4.8: Flow diagram for use case 5

6. Admin Add Organisation	
Summary	Admin adds organisation to the web application
Pre Conditions	Admin knows details for the new organisation. On admin login page
Primary Actors	Admin of the Web Application
Secondary Actors	System
Trigger	New organisation has signed up to use the application.
Flow of Events	<p>Admin: Logs into the application using a valid email and password.</p> <p>System: Brings admin to the admin home page</p> <p>Admin: Clicks Organisations in the navbar</p> <p>System: Brings admin to organisations page.</p> <p>Admin: Clicks add new organisation button</p> <p>System: Opens add organisation page.</p> <p>Admin: Enters all the detail of the new organisation.</p> <p>Admin: Clicks confirm button.</p> <p>System: Connects to Cloud Firestore and adds new organisation.</p> <p>System: Brings employee back to the organisations' page.</p>
Alternative Flow	<p>Admin: Cancels adding new organisation.</p> <p>System: Brings admin back to the organisations' page.</p> <p>OR</p> <p>System: Fails at adding new organisation</p> <p>System: Displays error message to admin</p>
Post Conditions	New Organisation is now registered and can now use the application.

TABLE 4.6: Use Case 6 - Admin Adds Organisation

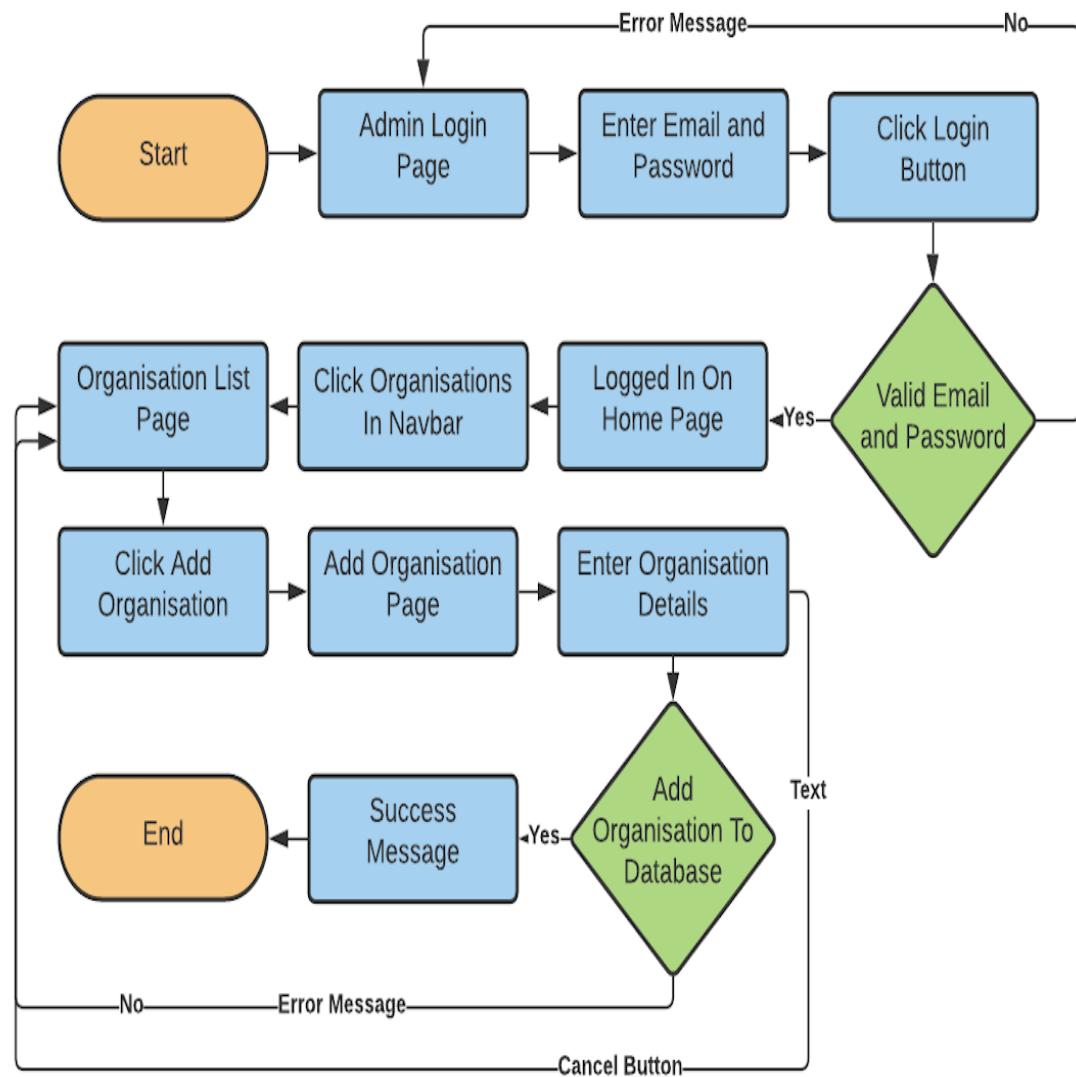


FIGURE 4.9: Flow diagram for use case 6

7. Documents AI	
Summary	Employee gets suggestions from AI
Pre Conditions	Employee is logged in, is on the add asset page and organisation has policy documents uploaded.
Primary Actors	Employee of Registered Organisation
Secondary Actors	System
Flow of Events	<p>Employee: Enters details of the asset</p> <p>System: Takes details of the asset and begins to scan for relevant policies</p> <p>System: Finds policies that might be relevant in the document.</p> <p>System: Displays the suggestions to the employee</p>
Alternative Flow	<p>System: Can't find any relevant documents.</p> <p>System: Displays message saying no documents found.</p>
Post Conditions	Suggestions are displayed to employee.

TABLE 4.7: Use Case 7 - AI Suggests Relevant Documents

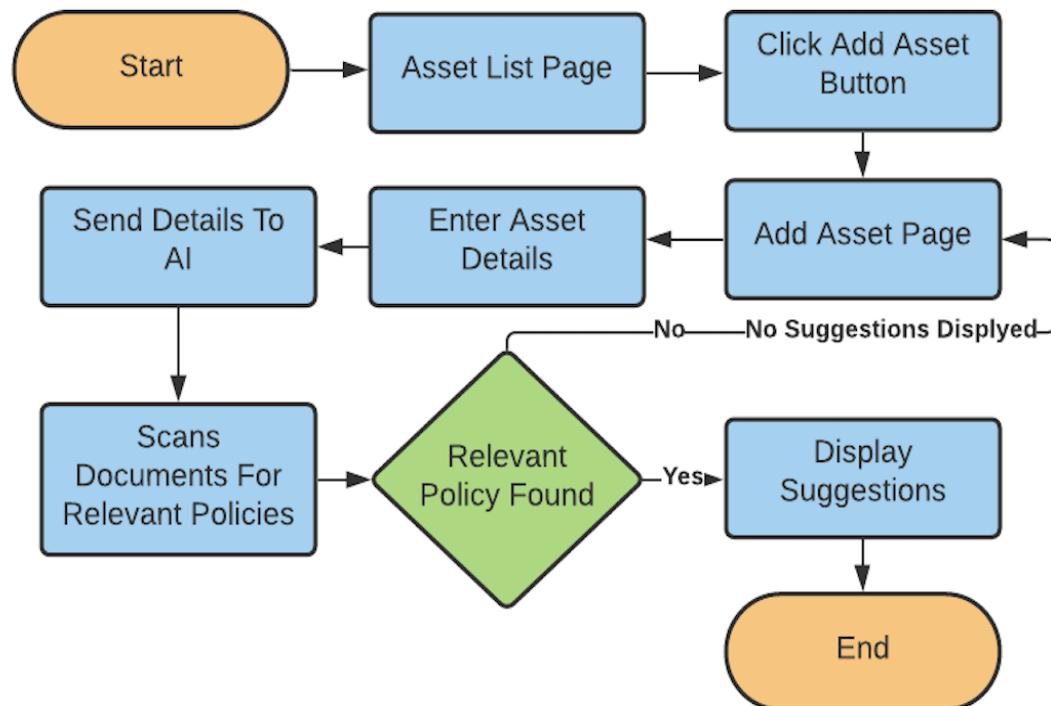


FIGURE 4.10: Flow diagram for use case 7

4.3 Risk Assessment

4.3.1 Risk 1

Risk	Issues with spaCy
Risk Probability/Consequence	Probable Major
Explanation	Lack of experience with spaCy could lead to issues with development.
Mitigation	Tutorials online will be followed along with a small practice project being completed before the main project.

TABLE 4.8: Potential Risk 1 - spaCy

4.3.2 Risk 2

Risk	Issues with Cloud Functions
Risk Probability/Consequence	Remote Minor
Explanation	Lack of experience with Cloud Functions could lead to issues with development.
Mitigation	Cloud Function documentation will be read and online tutorials will be watched/read and followed.

TABLE 4.9: Potential Risk 2 - Cloud Functions

4.3.3 Risk 3

Risk	Issues with Cloud Firestore
Risk Probability/Consequence	Remote Critical
Explanation	Lack of experience with Cloud Firestore could lead to issues with developing the database.
Mitigation	Cloud Firestore documentation will be read and online tutorials will be watched/read and followed.

TABLE 4.10: Potential Risk 3 - Cloud Firestore

4.3.4 Risk 4

Risk	Issues with Google Docs API
Risk Probability/Consequence	Remote Major
Explanation	Lack of experience with Google Docs API could lead to issues with development.
Mitigation	Google Docs API documentation will be read and a small practice project will be completed with spaCy before the main project.

TABLE 4.11: Potential Risk 4 - Google Docs API

TABLE 4.12: Initial risk matrix

Frequency/ Consequence	1-Rare	2-Remote	3-Occasional	4-Probable	5-Frequent
4-Fatal					
3-Critical		Risk 3			
2-Major		Risk 4		Risk 1	
1-Minor		Risk 2			

4.4 Methodology

To tackle the research needed for the background chapter of this project, websites like IEEE Xplore and Google Scholar were used to find relevant research that had already been completed in the areas relevant to this project.

In order to learn new technologies that will be needed for the project, different approaches will be used depending on the technology. There are 3 technologies used in this project that need to be learnt, spaCy, Google Docs API and Cloud Functions and Firestore.

- To learn more about spaCy, the tutorials available on the spaCy website will be completed along with making a small project to practice with.
- To learn more about Google Docs API the documentation available on the API's website will be read.
- To learn how to best use Cloud Functions and Firestore the documentation available on the Firebase website will be read along with watching tutorials available online.

To manage this project a Scrum approach will be used. A product backlog will be made at the beginning of the implementation phase to keep track of what needs to be done to develop the project. The implementation of this project will then be broken into 2-week sprints which will be planned out during sprint planning. Sprint planning will happen every two weeks at the beginning of the sprint.

4.5 Implementation Plan Schedule

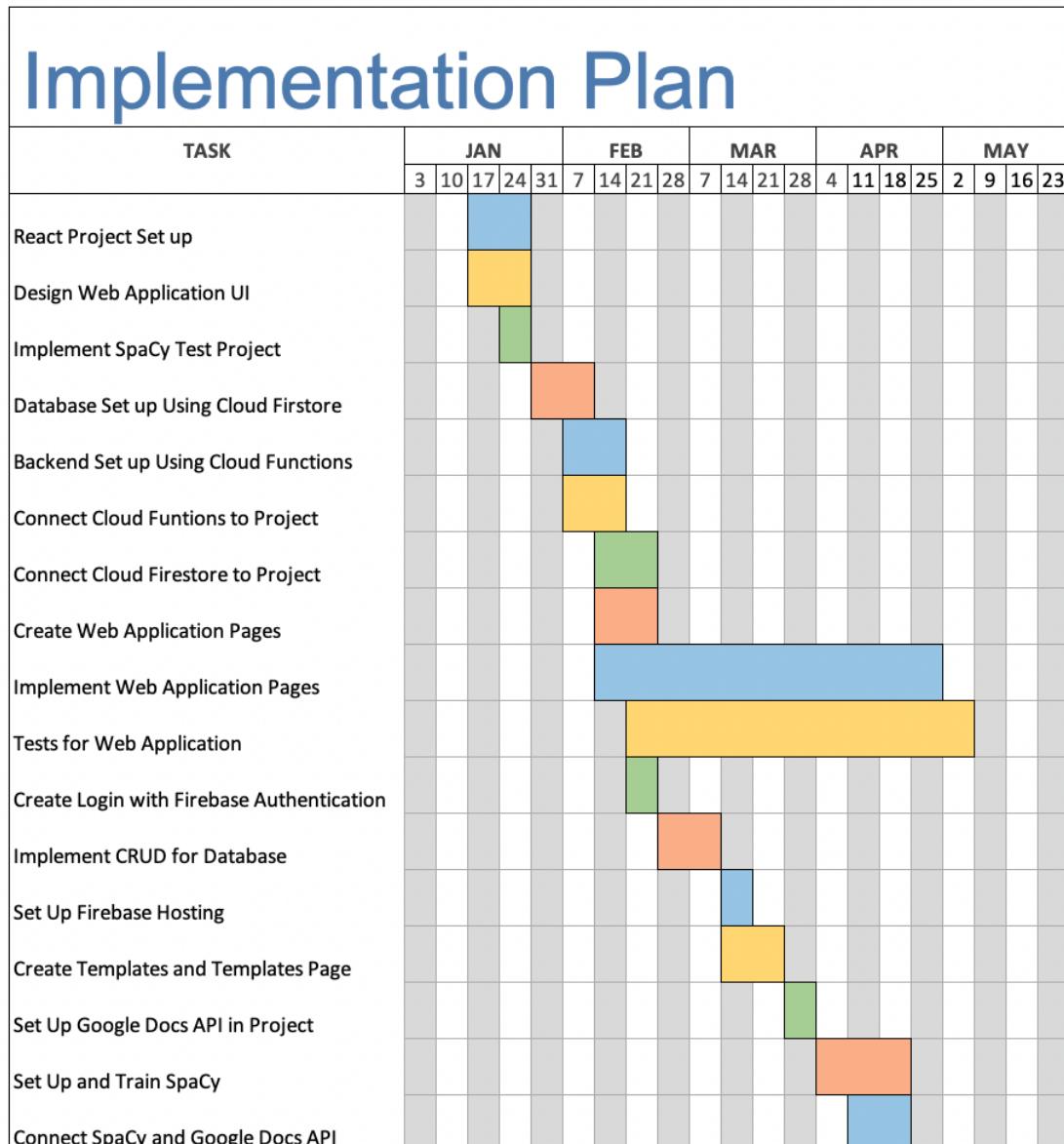


FIGURE 4.11: Implementation Plan for Semester 2

The above figure 4.11 shows a plan for the implementation of the project in semester 2. Each of the tasks will be further broken down into small tasks that will be completed during planned 2-week sprints. Trello^[35] will be used in order to plan and track the sprints.

4.6 Evaluation

To measure how much progress has been made with the goals of the project, a product backlog will be created by looking at the goals, functional requirements and implementation plan described in previous sections. This product backlog will contain multiple tasks that will be used in order to complete the goal and functional requirements of this project.

As mentioned in the implementation plan, 2-week sprints will be used during the development of this project. Each sprint will have a set of goals that need to be completed by the end of the sprint. These goals will be decided on at the beginning of each sprint by looking at the product backlog.

4.7 Wire Frames

This section will contain a visual representation of what the web application might look like when it is developed. Each of the pages will have a navbar to allow navigation throughout the application along with a footer and header.

Figure 4.12 shows what the home page that the user will be brought to once they log in will look like, it will contain some stats on the user assets along with any activity on those assets and notifications.

Figure 4.13 shows what the page that lists all the assets will look like, it will have a table which contains all the assets and an add asset button. This controls list page will look the same as the asset list page just containing controls instead of assets.

Figure 4.14 shows what the asset page will look like once a user clicks on an asset in the asset list table. This page will have all the details of the asset displayed including the asset name, type, description, owner details, the associated risks and controls along with a review, edit and delete button.

Figure 4.15 shows what the risk page will look like after a user clicks on a risk on an asset or control page. The page will contain the risks details including the risks name, severity, status, description, owner details of associated asset, a list of associated controls along with an edit and delete button.

Figure 4.16 shows what the control page will look like when a user clicks on it through a risk page or controls list page. The page will contain all the control detail including the control name, type, description, the risks controlled along with an edit and delete button.

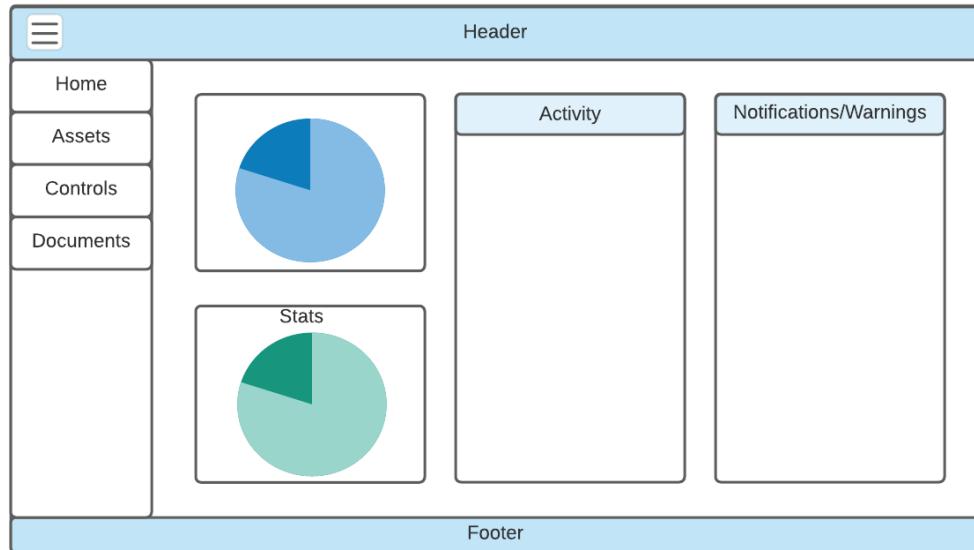


FIGURE 4.12: Wire Frame for Home Page of the application



FIGURE 4.13: Wire Frame for page to display list of all the assets

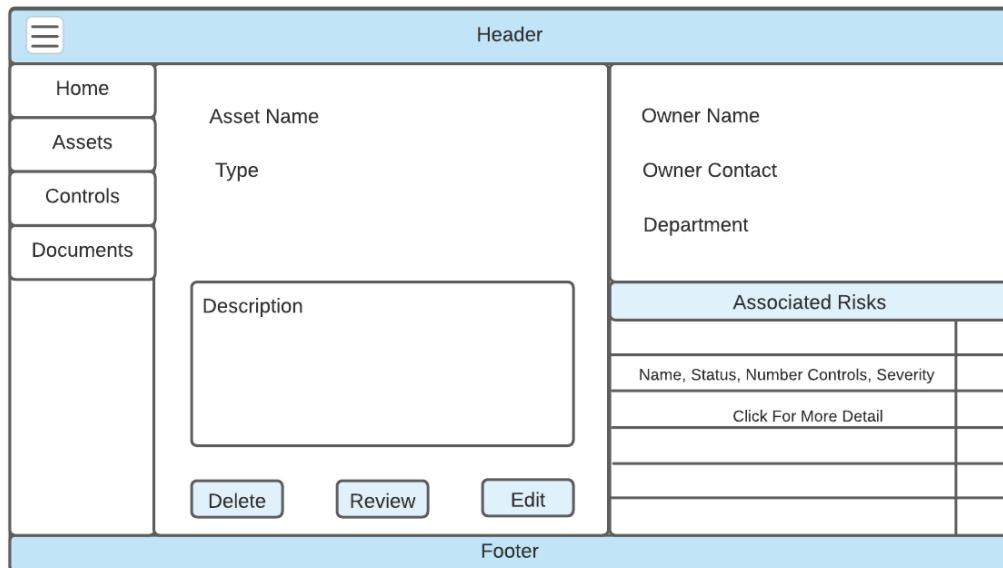


FIGURE 4.14: Wire Frame for Asset Page where detail of one asset is displayed

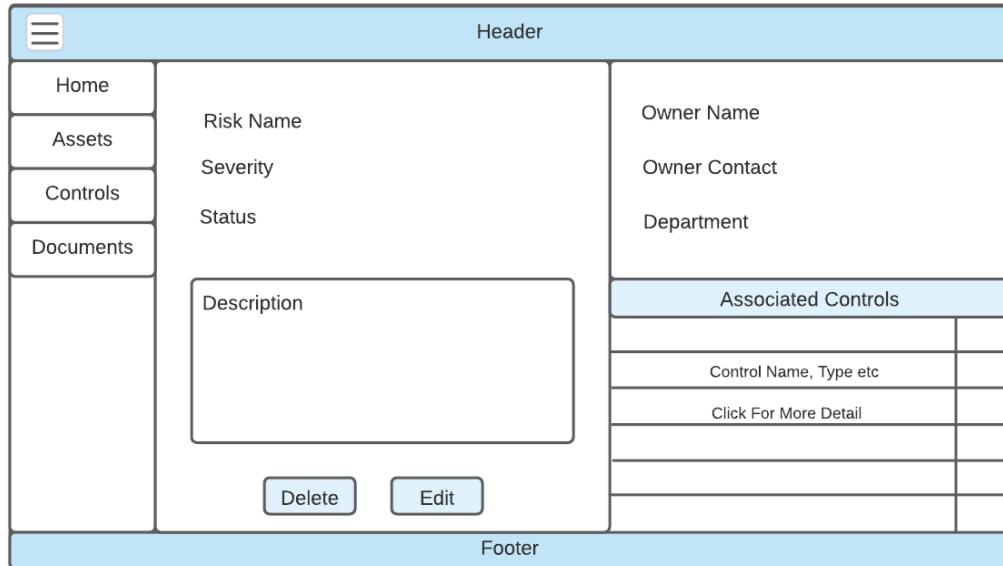


FIGURE 4.15: Wire Frame for Risk Page where detail of one risk is displayed

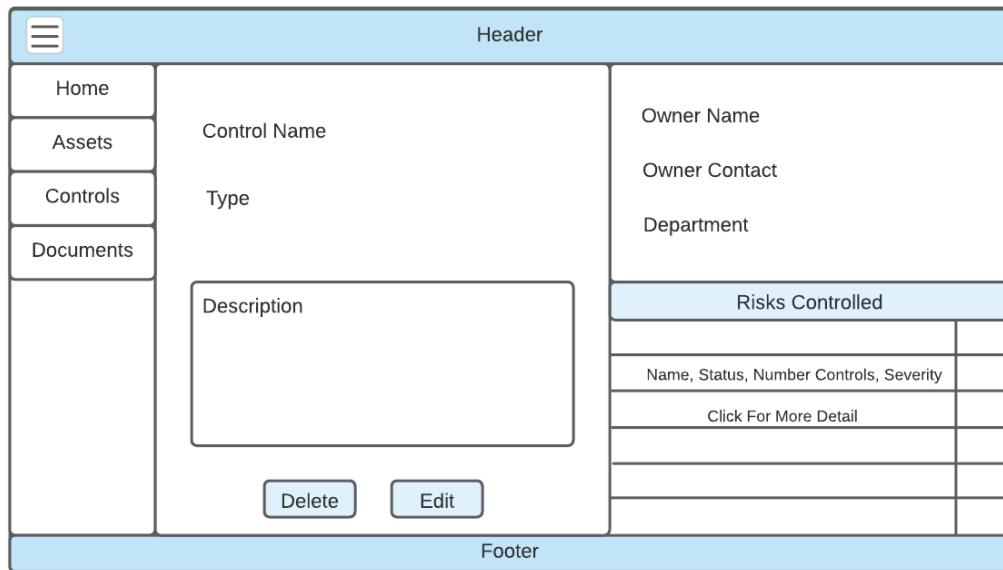


FIGURE 4.16: Wire Frame for Control Page where detail of one control is displayed

Chapter 5

Implementation

5.1 Difficulties Encountered

Number	Difficulty	Level
1	Firebase Version 9 Changes	Medium
2	TypeScript to JavaScript	Medium
3	React Router Changes	Easy
4	Sign In Issue	Medium
5	Storing User Data with UID	Easy
6	On Row Click	Easy
7	Time Constraints	Hard

TABLE 5.1: Difficulties Encountered Categories

5.1.1 Difficulty 1

Title: Firebase Version 9 Changes

Affected Area: Database/Web Application

Description of Difficulty: A few months before the implementation of this project was started, Firebase released version 9 which had massive changes compared to version 8. Firebase was switched to a modular, functional programming style to make the Firebase smaller. This made Firebase faster and load quicker. But this also meant that the way in which Firebase was used within an application was changed. The imports and programming styles were changed which meant previous knowledge and experience with Firebase was no longer helpful[36].

Solution: Reading through Firebase V9 documentation and guides on switching to V9 from V8 solved the issues.

5.1.2 Difficulty 2

Title: Switch from TypeScript to JavaScript

Affected Area: Web Application

Description of Difficulty: 6 weeks into the implementation of this project after encountering countless issues with trying to implement Firebase authentication for user login and connecting to Firestore. The decision to switch from TypeScript to JavaScript was made, this was due to there being more information and help on using JavaScript with the new Firebase version 9.

Solution: Switching a project from TypeScript to JavaScript was time-consuming but worth it in the end. A copy of the project was made and all files were renamed and slight code changes had to be made. A new GitHub was then made to store the new project.

5.1.3 Difficulty 3

Title: React Router Changes

Affected Area: Web Application

Description of Difficulty: During the implementation of the routing it was realised that there had been an update to react-router. This meant that the previous way in which react-router was implemented no longer worked and needed to be updated in order to work.

Solution: Following an online tutorial instead of relying on past experience solved the issues.

5.1.4 Difficulty 4

Title: Sign In Issue

Affected Area: Web Application

Description of Difficulty: During the implementation of the sign out button it was discovered that the sign-in functionality wasn't working as expected. It was realised that reloading the page, sign out and routing were broken due to the user being null when they had signed in. This caused the user to be unable to sign out and routing get confused when the page was reloaded as it didn't know where to send the user.

Solution: After a few days of looking for a solution one was found which was to make a function to check if it was loading and then if it wasn't loading to look for the user and check if they were logged in.

5.1.5 Difficulty 5

Title: Storing User Data with UID

Affected Area: Web Application/Database

Description of Difficulty: When trying to store user data in the database after getting more details about the user, a problem was encountered with trying to store the user with the user's unique ID generated by Firebase Authentication instead of a new unique ID generated by Firestore.

Solution: This was solved easily enough after eventually finding some instructions online about how to get the user's unique ID and send it to the database.

5.1.6 Difficulty 6

Title: Sending Data on Row Click with Data Grid

Affected Area: Web Application

Description of Difficulty: The project uses MUI Data Grid which displays a table of data to the user. Issues were encountered when trying to implement sending a row of data to a new page when a row in the table was clicked on. Trying to route the user to a new page along with the data proved more challenging than expected as the ways suggested on the MUI website didn't work for this project.

Solution: This was solved eventually by creating an action button within the data grid allowing the user to click on the arrow button and be brought to the page with more detail about the asset.

5.1.7 Difficulty 7

Title: Time Constraints

Affected Area: Entire Project

Description of Difficulty: The level of work was underestimated not only for this project but also for other modules. This led there to there being way less time for this project than expected and certain functionality having to be dropped.

Solution: There was no solution to this issue.

5.2 Actual Solution Approach

The original plan for this project was to include an AI but after discussion with the project supervisor, a decision was made to drop the AI as it overcomplicated the system and there was a more efficient way to provide the same functionality. Instead of AI which would require a lot of time as the system would need to be working a long time in order to learn from the users, tagging was instead implemented. The tagging system provides the same functionality but took less time and therefore allowed more time to be focused on the web applications' other functionalities which were the main focus of this project.

The changes that had to be made because of this are explained in this section along with other changes that needed to be made.

5.2.1 Architecture of the Solution

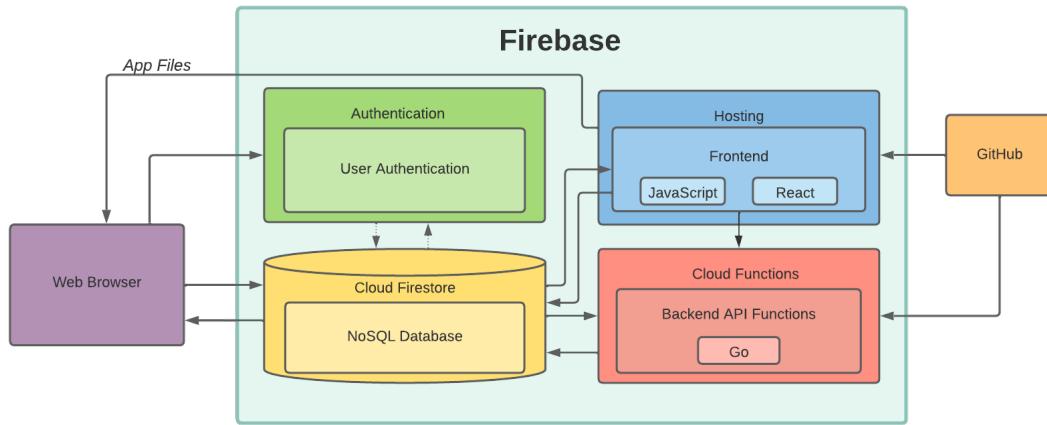


FIGURE 5.1: High Level Architecture Diagram for Web Application Solution

There were only 2 changes between the architecture that was made during the original design and the solution for the final product. The first change was the removal of the AI portion of the architecture as the AI was removed from the project. This meant both the Google Docs API and spaCy were removed from the diagram as they were contained in the AI section of the diagram.

The second change was the switch from TypeScript to JavaScript in the frontend section of the architecture diagram.

5.2.2 Technologies Used

Below is an updated list of the technology which was used to develop the web application for this project.

- **React** - JavaScript library used to build the frontend application.
- **JavaScript** - The programming language used to build the frontend application.
- **MUI Core** - React component library used to build the frontend application.
- **Firebase Hosting** - Used to host the web application.
- **Cloud Firestore** - A cloud based, NoSQL database used as the database for this application.
- **Firebase Authentication** - Used to authenticate users in the application.
- **Cloud Functions** - Used for parts of the backend of the application.
- **Go** - The programming language used to write the Cloud Functions.
- **Git Hub** - Used as the version control for this project.
- **Visual Studio Code** - The IDE that was used to develop this application.

5.2.3 Use Cases

There were a few changes with use cases due to dropping the AI and the way the application was set up. Use cases 6 and 7 outlined in the chapter previous to this one no longer apply to the project. Since the database will be individual to each organisation there is no need to have the ability to add an organisation so use case 6 is no longer needed. Use case 7 is no longer needed as it was about the AI which was dropped in place of a tagging system. A new use case outlined below details a use case for the tagging system.

8. Tagging System	
Summary	Employee gets suggestions from Tagging System
Pre Conditions	Employee is logged in, is on the add asset page there are risks or controls on the system
Primary Actors	Employee of Registered Organisation
Secondary Actors	System
Flow of Events	Employee: Enters details of the asset or risk Employee: Enters tags for the asset System: Takes tags of the asset and filters all risks or controls for relevant risk or controls System: Displays the suggestions to the employee
Alternative Flow	System: Can't find any relevant risks or controls. System: Displays empty list.
Post Conditions	Suggestions are displayed to employee.

TABLE 5.2: Use Case 8 - Tagging System

5.2.4 Risk Assessment

The risk assessment carried out in the research phase identified four risks. Two of these risks are no longer relevant as the AI aspect was removed from the project. The other two identified risks are still relevant and did occur during the implementation of this project.

5.2.4.1 Risk 2

Risk	Issues with Cloud Functions
Risk Probability/Consequence	Remote Minor
Encountered	Yes
Explanation	Lack of experience with Cloud Functions could lead to issues with development.
Mitigation	Cloud Function documentation will be read and online tutorials will be watched/read and followed.

TABLE 5.3: Potential Risk 2 - Cloud Functions

5.2.4.2 Risk 3

Risk	Issues with Cloud Firestore
Risk Probability/Consequence	Remote Critical
Encountered	Yes
Explanation	Lack of experience with Cloud Firestore could lead to issues with developing the database.
	Cloud Firestore documentation will be read and online tutorials will be watched/read and followed.

TABLE 5.4: Potential Risk 3 - Cloud Firestore

5.2.5 Methodology to Develop Solution Approach

To manage the implementation of this project a Scrum approach was used. There was a product backlog made at the beginning of the implementation phase, and the project implementation was broken into 2-week sprints. At the start of each sprint, a new Trello board was made for the sprint, tasks were then taken from the product backlog and put in the Trello board along with any tasks left over from the previous sprint.

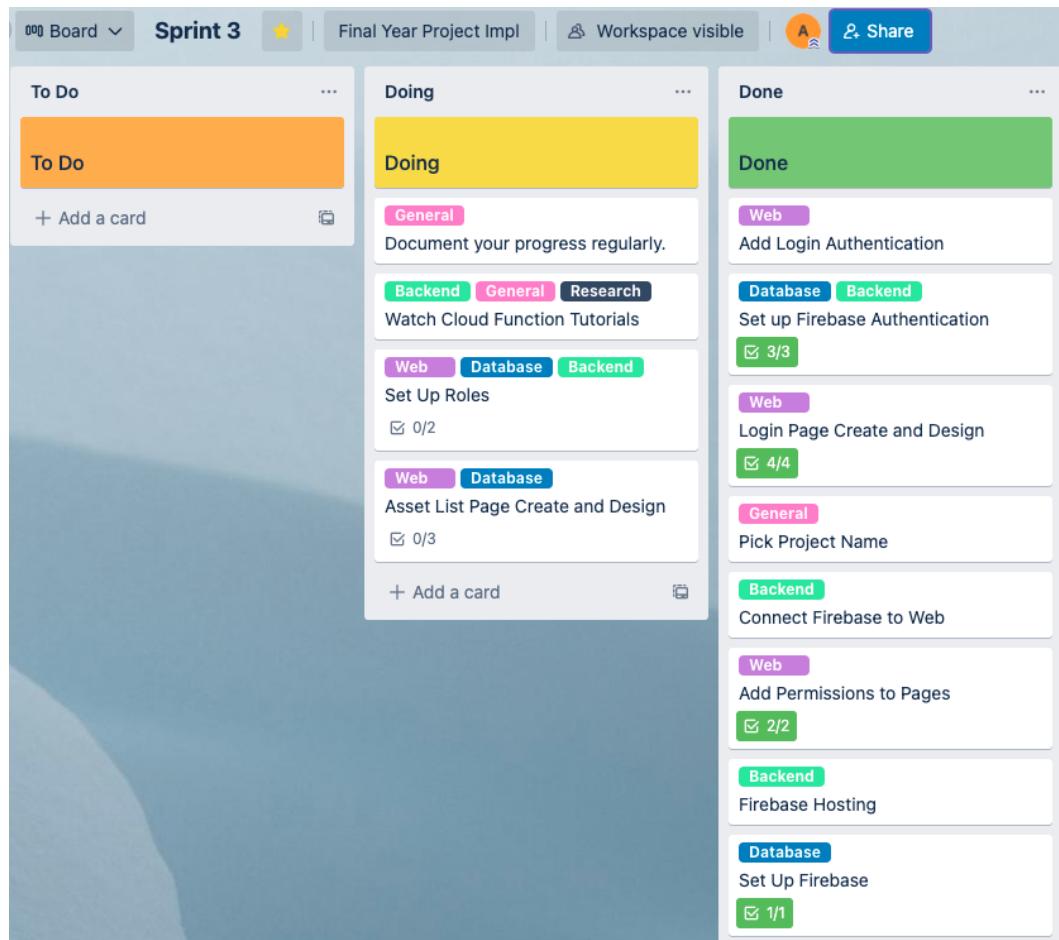


FIGURE 5.2: Trello Board for Sprint

5.2.6 Implementation Schedule

The implementation schedule changed a lot from the research phase. When originally making the plan, the level of work needed for not only aspects of this project but also other modules was underestimated. The removal of the AI at the beginning of this phase allowed for more time to be spent on the web pages but still, time constraints caused the actual implementation to differ from the original plan.

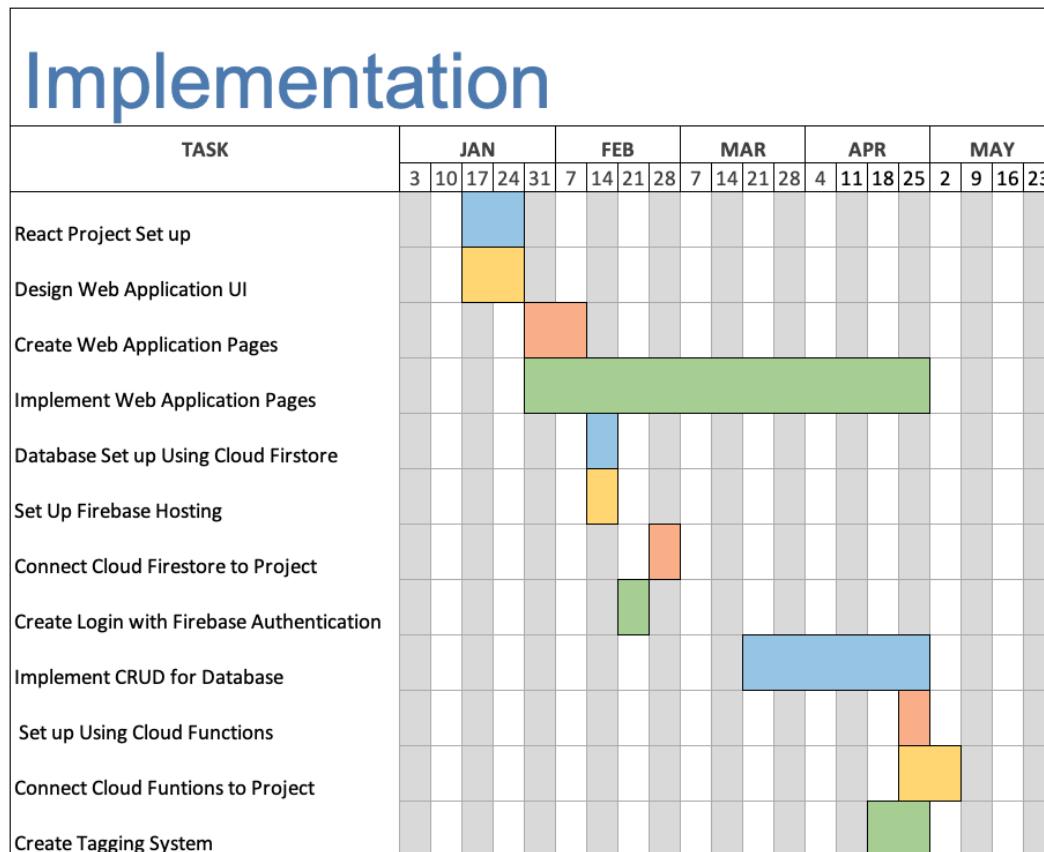


FIGURE 5.3: Implementation Chart

5.2.7 Evaluation

The evaluation for this project didn't change from the plan made in the research phase of this project. A backlog was made at the beginning of the implementation phase and tasks were taken for each sprint. Sometimes due to tasks being blocked by an issue that was encountered in the previous sprint items had to be moved to the next sprint. The backlog is shown below it as the tasks were completed they were marked with a done, the tasks were ordered by what needed to be done before the next could be done or by which task too priority for the project.

	Task	Description/Acceptance Criteria	Status
Frontend Web	Create Blank Pages	Blank pages for every page for routing setup	Done
Frontend Web	Colour Pallette	Pick colour pallette	Done
Frontend Web	Routing of Pages	Routing between pages using nav basic	Done
General	Pick Project Name	Name Chosen	Done
General	Set Up Project	Project Basic Set Up	Done
Frontend Web	Nav Bar Basic Set Up	Nav Component Created basic	Done
Database	Design Database	Design Main Database	Done
General	Firebase Account Set Up	Set Up Firebase Account	Done
Backend Web	Database Connection	Connect Database To web	Done
Frontend Web	Nav Bar UI/UX Set UP	Make Nav more user friendly and comeplete	Done
Frontend Web	Login Page	Page created and complete	Done
Backend Web	Login Set Up	Add Login Functionality	Done
Database	User Database	Set Up Database	Done
Database	Database Set Up	Set Up Database	Done
Backend Web	Authentication Set Up	Login and Sign up Functionality	Done
Frontend Web	Sign Up Page	Page created and complete	Done
Backend Web	Watch Cloud Functions Tutorials	Watch Cloud Function Tutorials	Done
Database	CRUD For Assets	CRUD created and connected	Done
Frontend Web	Asset Page List	Page created and complete	Done
Frontend Web	Controls Page List	Page created and complete	Done
Database	CRUD For Controls	CRUD created and connected	Done
Frontend Web	Risk Page	Page created and complete with popup	Done
Database	CRUD For Risks	CRUD created and connected	Done
Frontend Web	Control Page	Page created and complete	Done
Frontend Web	Asset Page	Page created and complete	Done
Frontend Web	Add Control Page	Page created and complete	Done
Frontend Web	Add Control Page	Page created and complete	Done
Frontend Web	Add Asset Page	Page created and complete	Done
Tag	Add Tagging Ssystem to Database	Tagging Filter Working	Done
Tag	Add Tagging to Add Asset and Risk	Tags Displayed on Page	Done
Backend Web	Set Up Cloud Functions	Create Functions	Done
Backend Web	Connect Cloud Functions to Web	Connect Functions and Web	Done
Frontend Web	Validation	All Inputs Validated	To Do
Frontend Web	Google Docs Tagging	Google Docs are Tagged	To Do
Frontend Web	Forgot Password Web Page	User Can Reset Password	To Do
Frontend Web	Add User Page For Admin	User Can Be Added By Admin In App	To Do

FIGURE 5.4: Project Backlog

5.2.8 Prototype of Resulting Product

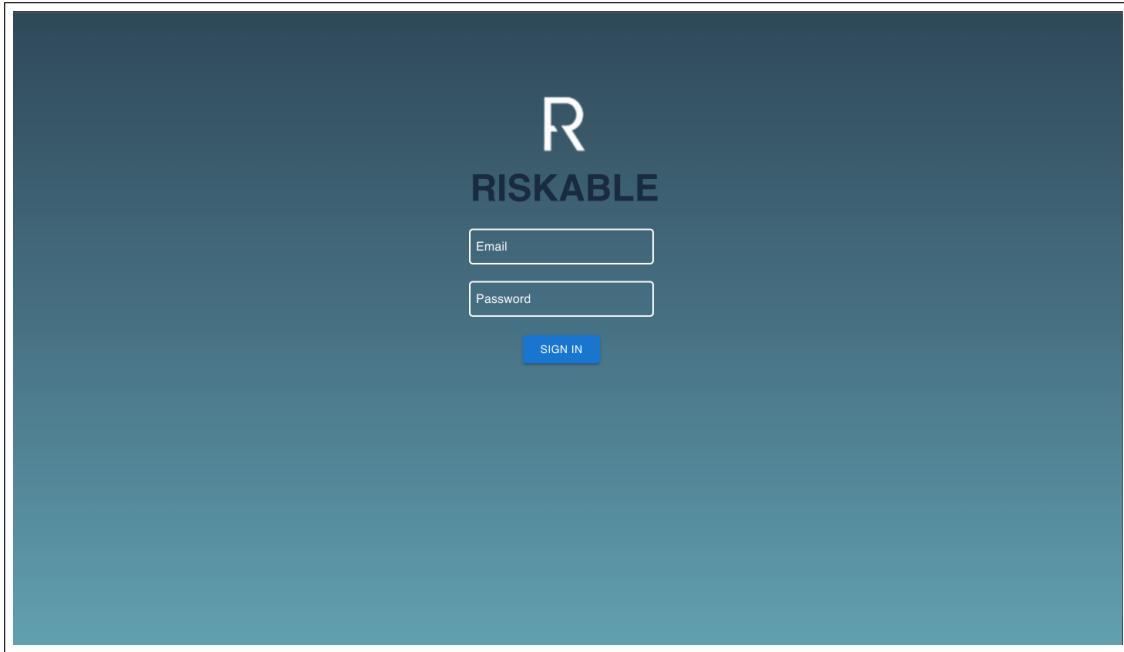


FIGURE 5.5: Login Page

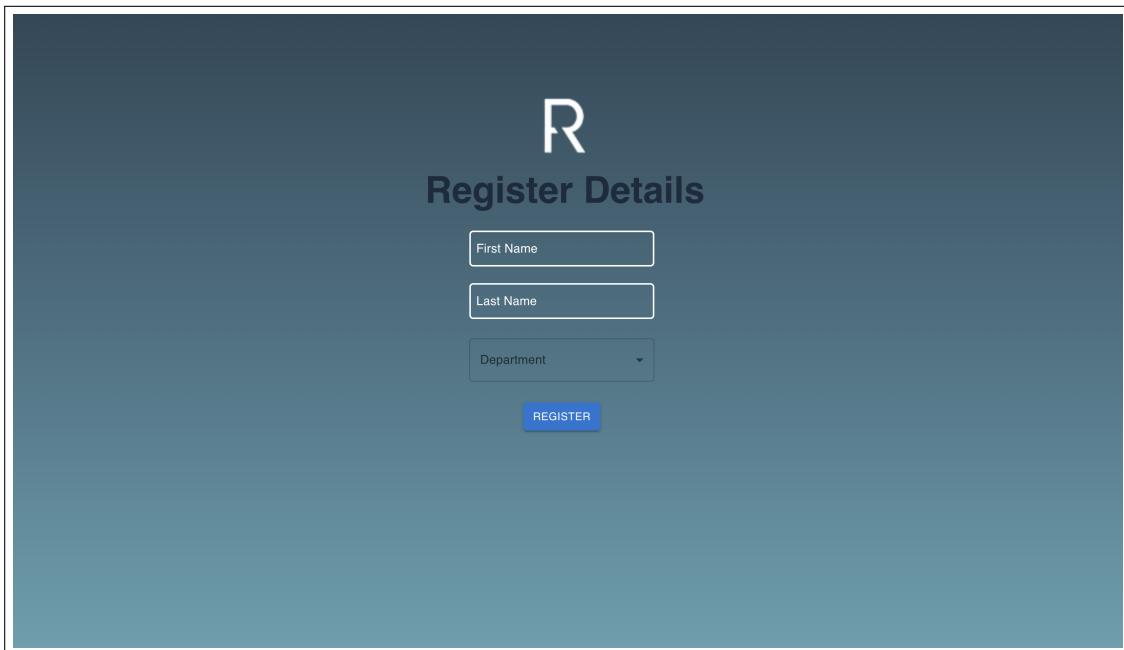


FIGURE 5.6: Register Details Page

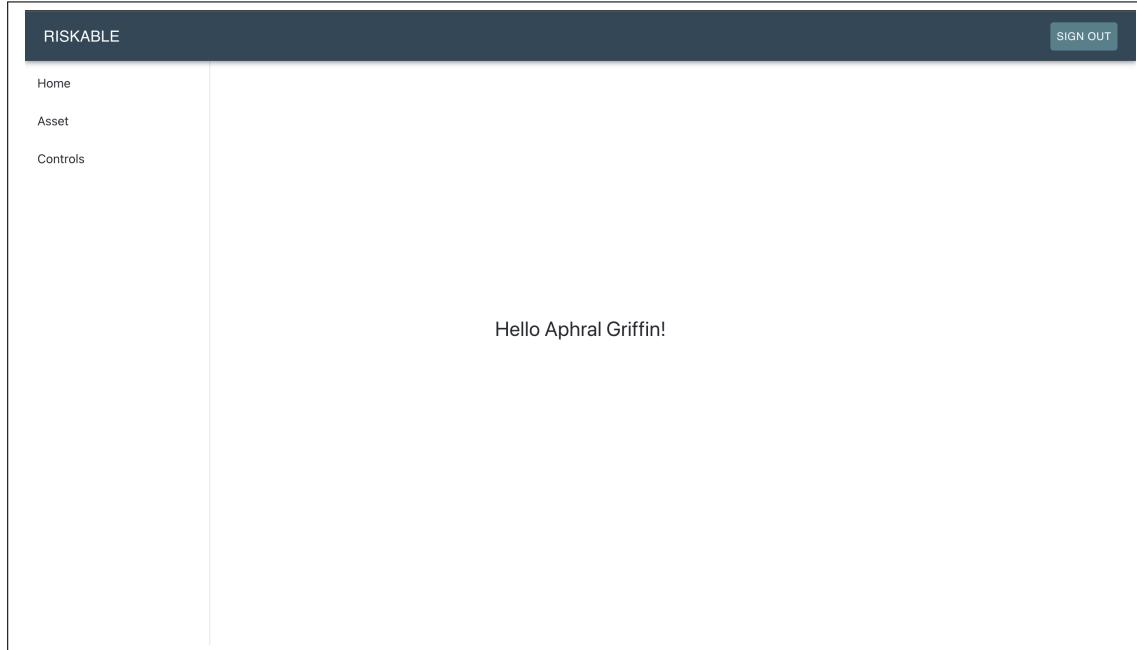


FIGURE 5.7: Home Page

The screenshot shows the RISKABLE application's asset list page. At the top, there is a dark header bar with the word "RISKABLE" on the left and a "SIGN OUT" button on the right. Below the header, a vertical sidebar on the left contains three menu items: "Home", "Asset", and "Controls". The main content area on the right has a title "Assets" centered at the top. It features a search bar labeled "Filter" and a "ADD ASSET" button. Below the search bar is a table with six rows of asset data. The table columns are "Name", "Type", "Department", and "Actions". The assets listed are: Customer Data (Data, Sales), Laptops (Equipment, Engineering), Building (Building, Management), Warehouse (Building, Warehouse), and Sales Leads (Data, Sales). At the bottom of the table, there is a page navigation indicator showing "1–5 of 7" with arrows for " < " and " > ".

Name	Type	Department	Actions
Customer Data	Data	Sales	→
Laptops	Equipment	Engineering	→
Building	Building	Management	→
Warehouse	Building	Warehouse	→
Sales Leads	Data	Sales	→

FIGURE 5.8: Asset List Page

The screenshot shows the 'Asset Details' page for an asset named 'Customer Data'. The asset has an ID of '7g5hoAE9cjVzpoJhoCic' and is categorized as 'Data'. It was reviewed on '24/04/2022'. The description is 'Sales customers personal data'. On the right, there is a section for 'Owner Details' showing 'Aphral Griffin' from the 'Engineering' department, with contact email 'aphral.griffin+1@gmail.com'. Below this is a table titled 'Asset Risks' with one entry: 'Loss of Data' (Severity: High, Status: Medium). Navigation buttons at the bottom include 'DELETE', 'EDIT', 'REVIEW', and 'ADD RISK'.

FIGURE 5.9: Asset Detail Page

The screenshot shows the 'Update Asset' page. The title is 'Update Asset: Customer Data'. The form contains fields for 'Asset Name' (Customer Data), 'Asset Type' (Data), 'Asset Description' (Sales customers personal data), and 'Asset Department' (Sales). At the bottom are 'SAVE' and 'CANCEL' buttons.

FIGURE 5.10: Update Asset Page

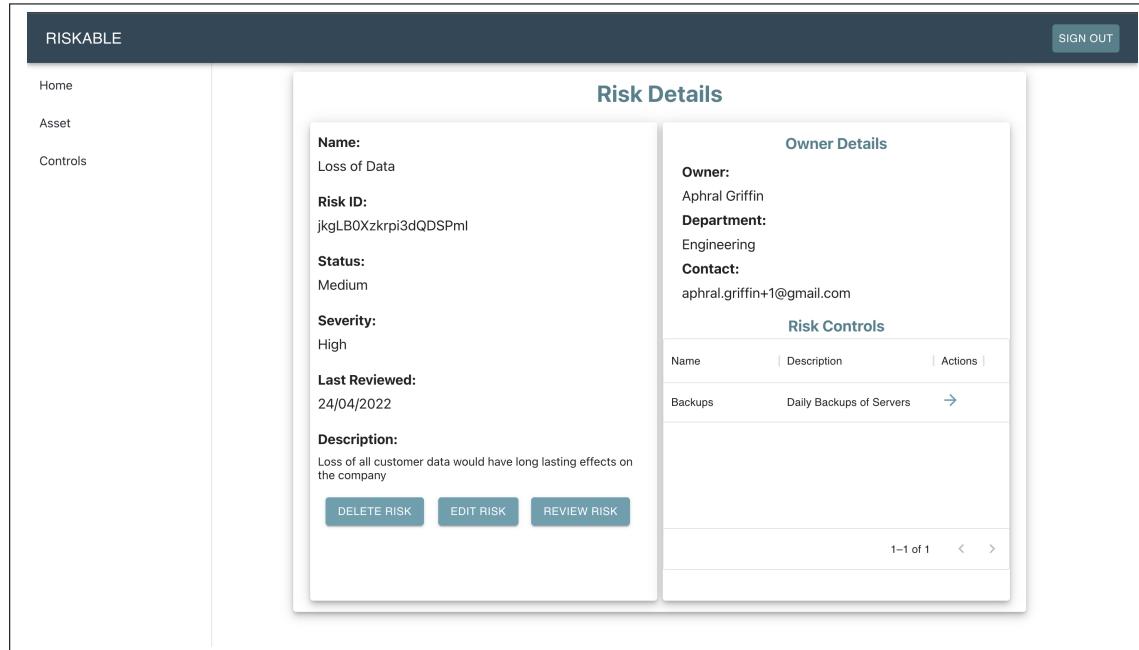


FIGURE 5.11: Risk Detail Page

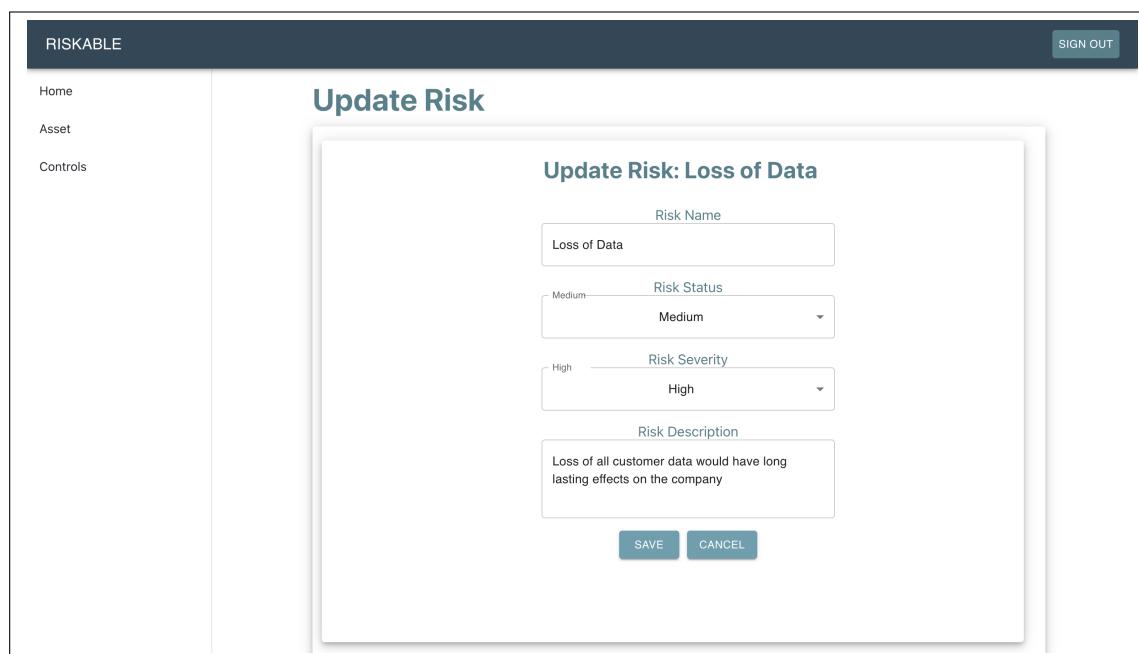


FIGURE 5.12: Update Risk Page

The screenshot shows the 'Controls' page of the RISKABLE application. The left sidebar has links for Home, Asset, and Controls. The main area is titled 'Controls' and contains a table with two rows. The table columns are 'Name', 'Reviewed', 'Description', and 'Actions'. The first row shows 'Backups' with 'Reviewed' status and 'Daily Backups of Servers' description. The second row shows 'Fire Policy' with 'Reviewed' status and '24/04/2022' date, and a description stating 'Fire Policy is observed and followed. Full document available he...'. A 'Filter' input field and an 'ADD CONTROL' button are at the top of the table. A pagination bar at the bottom indicates '1–2 of 2'.

FIGURE 5.13: Control List Page

The screenshot shows the 'Add Control' page of the RISKABLE application. The left sidebar has links for Home, Asset, and Controls. The main area is titled 'Add Control' and contains a form titled 'Add Control Form'. The form has three input fields: 'Control Name *', 'Control Description *', and 'Tags To Label Control'. At the bottom are 'SAVE' and 'CANCEL' buttons.

FIGURE 5.14: Add Control Page

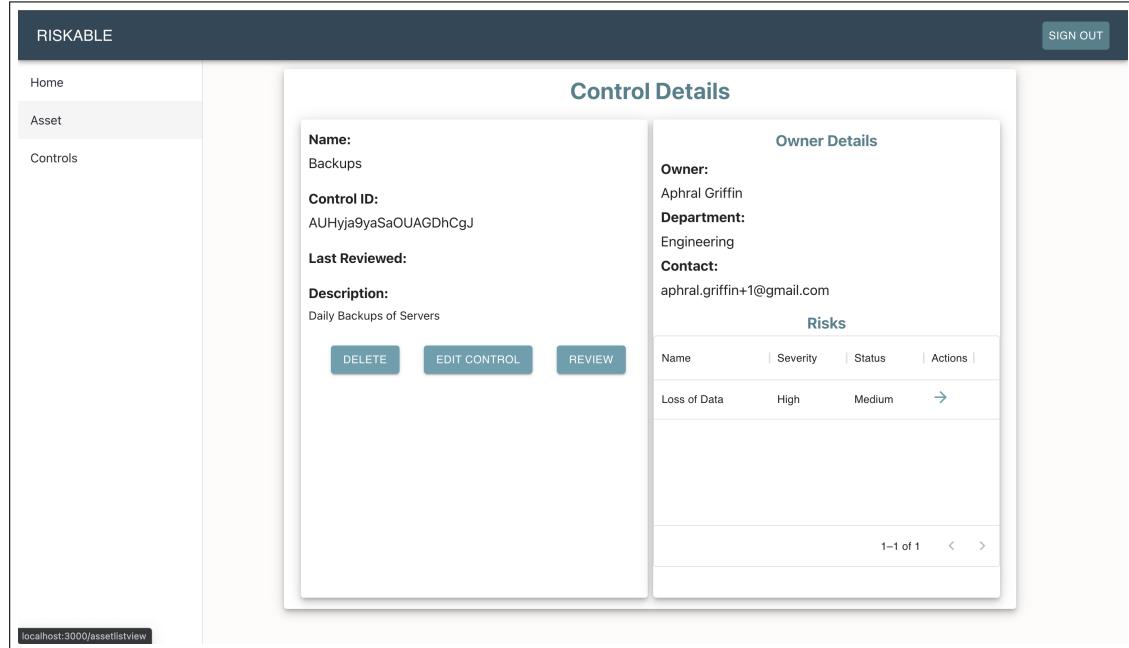


FIGURE 5.15: Control Details Page

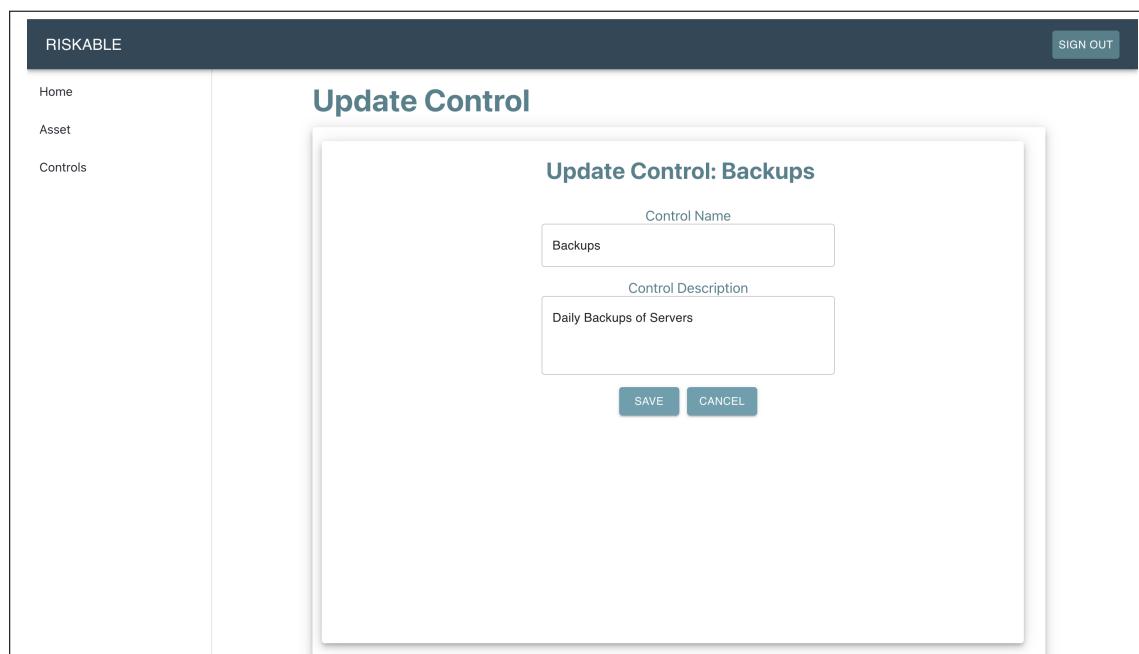


FIGURE 5.16: Update Control Page

The screenshot shows the 'Add Asset' page. On the left, a sidebar menu includes 'Home', 'Asset', and 'Controls'. The main area has a title 'Add Asset' and a sub-section 'Add Asset Form' containing fields for 'Asset Name*', 'Asset Associated Department*', 'Asset Type*', 'Asset Description*', and 'Tags To Label Asset' (with a placeholder 'Data'). Below these are 'SAVE' and 'CANCEL' buttons. To the right is a 'Risk Suggestions' section listing four items: 'Name' (Tags), 'Inspection' (Building,Fire), 'Backup' (Data), and 'Training' (Data,Email). A pagination indicator '1–3 of 3' with arrows is at the bottom of this section.

FIGURE 5.17: Add Asset Page

This screenshot shows the same 'Add Asset' page after a tag has been added. In the 'Tags To Label Asset' field, 'Data' is entered and highlighted with a red border. The 'Risk Suggestions' section now shows only two items: 'Backup' (Data) and 'Training' (Data,Email). The pagination indicator '1–2 of 2' is visible at the bottom.

FIGURE 5.18: Add Asset Page Where Tag is Added and Data is Filtered

The screenshot shows the 'Add Risk' page of a web application. The top navigation bar has a dark blue header with the word 'RISKABLE' on the left and a 'SIGN OUT' button on the right. On the far left, there is a vertical sidebar with three menu items: 'Home', 'Asset', and 'Controls'. The main content area is titled 'Add Risk' at the top. It contains a sub-section titled 'Add Risk Form' which includes fields for 'Risk Name*', 'Risk Status', 'Risk Severity', 'Risk Description*', and 'Tags To Label Risk'. Below these fields are two buttons: 'SAVE' and 'CANCEL'. To the right of the form is a section titled 'Control Suggestions' which displays a table with two rows. The first row shows 'Name' and 'Tags' with entries 'Backups' and 'Data'. The second row shows 'Fire Policy' and 'Fire,Building'. At the bottom of the right panel, there is a pagination indicator '1-2 of 2' with arrows.

FIGURE 5.19: Add Risk Page

This screenshot shows the same 'Add Risk' page as Figure 5.19, but with a visible change in the 'Tags To Label Risk' field. The field now contains the text 'Fire' followed by a small circular icon with an 'X' and a delete button 'x'. The rest of the interface remains identical to Figure 5.19, including the 'Control Suggestions' panel which still lists 'Backups' and 'Data' under 'Name' and 'Tags' respectively.

FIGURE 5.20: Add Risk Page Where Tag is Added and Data is Filtered

Chapter 6

Testing and Evaluation

6.1 Use Case Testing - Functional Requirements

Each use case that is still relevant to the project was carried out in order to test the web application. The results of each test are detailed below along with corresponding screenshots for each use case.

6.1.1 Use Case 1 - Login

Title: Login		
Main Success Scenario	Step	Description
E: Employee S: System	1	E: Enters Username and Password, clicks Login button.
	2	S: Checks Employee details are correct
	3	S: Brings Employee to home page.
Extensions	2a	Username or Password Not Valid S: Display Error Message

TABLE 6.1: Use Case 1 - Login

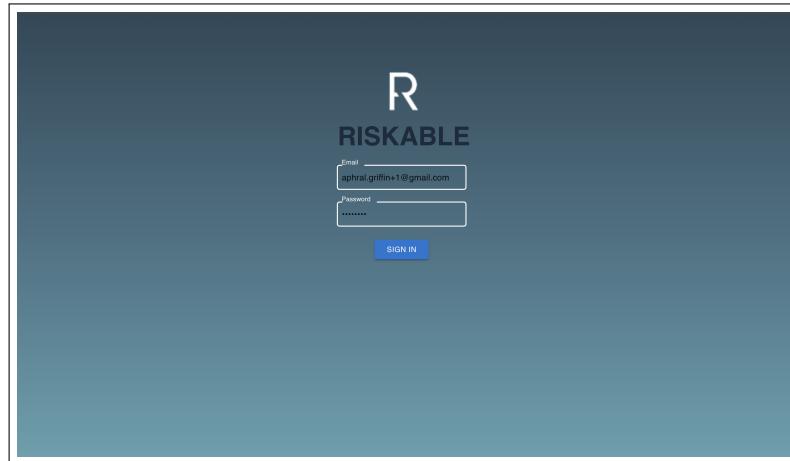


FIGURE 6.1: User Enters Valid Username and Password and Clicks Login - Step 1

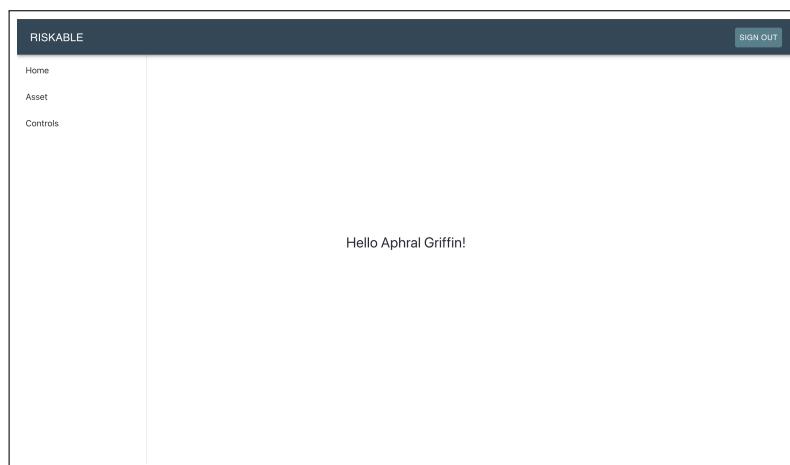


FIGURE 6.2: User on Home Page after Login - Step 2 & 3

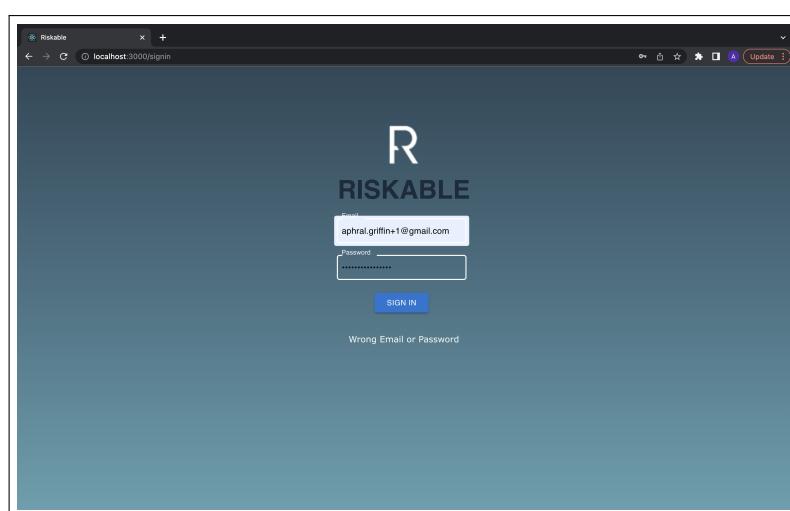


FIGURE 6.3: Error Message Displayed after Wrong Username or Password - Step 2a

6.1.2 Use Case 2 - Maintain Asset

Title: Add Asset		
Main Success Scenario	Step	Description
E: Employee S: System	1	E: Fills in Asset Details & Selects Relevant Risks
	2	E: Clicks Add Asset Button
	3	S: Adds Asset To Database
	4	S: Displays Success Message
	5	S: Brings Employee to Asset List Page
Extensions	2a	Cancel Button Clicked S: Brings Employee to Asset List Page
	3a	Error Adding Asset S: Display Error Message

TABLE 6.2: Use Case 2 - Maintain Asset - Add Asset

Title: Update Asset		
Main Success Scenario	Step	Description
E: Employee S: System	1	S: Populates Fields with Asset Details
	2	E: Fills in Asset Details
	3	E: Clicks Update Asset Button
	4	S: Updates Asset In Database
	5	S: Displays Success Message
	6	S: Brings Employee to Asset Details Page
Extensions	2a	Cancel Button Clicked S: Brings Employee to Asset Details Page
	3a	Error Updating Asset S: Display Error Message

TABLE 6.3: Use Case 2 - Maintain Asset - Update Asset

Title: Delete Asset		
Main Success Scenario	Step	Description
E: Employee S: System	1	E: Clicks Delete Button
	2	S: Displays Warning asking Employee are They Sure
	3	E: Clicks Confirm
	4	S: Deletes Asset From Database
	5	S: Displays Success Message
	6	S: Brings Employee to Asset List Page
Extensions		Cancel Button Clicked
	2a	S: Brings Employee to Asset Details Page
	4a	Error Updating Asset S: Display Error Message

TABLE 6.4: Use Case 2 - Maintain Asset - Delete Asset

The screenshot shows the 'Add Asset' form in the RISKABLE application. The form has fields for Asset Name, Asset Associated Department, Asset Type, Asset Description, and Tag To Label Asset. A success message 'localhost:3000 says Success Adding Asset' is displayed at the top right.

FIGURE 6.4: User Successfully Add Asset - Adds Asset - Step 1 - 4

The screenshot shows the 'Assets' list page in the RISKABLE application. The table lists assets with their details and actions. The assets listed are Customer Data, Laptops, Building, Warehouse, and Sales Leads.

FIGURE 6.5: Cancel Button Clicked - Add Asset - 2a

The screenshot shows the 'Add Asset' form. The 'Asset Name' field contains 'Testing Web Add'. The 'Asset Associated Department' field contains 'Sales'. The 'Asset Type' field contains 'Data'. The 'Asset Description' field is empty. Below the form, a message box displays the error: 'Error: Problems Adding Asset'. At the bottom right of the message box are 'SAVE' and 'CANCEL' buttons.

FIGURE 6.6: Error Adding Asset - Add Asset - 3a

The screenshot shows the 'Update Asset: Testing Asset' form. The 'Asset Name' field contains 'Testing Asset Edit'. The 'Asset Type' field contains 'Data'. The 'Asset Description' field contains 'Testing'. The 'Asset Department' field contains 'Sales'. A success message at the top of the page reads 'localhost:3000 says Success Updating Asset'. At the bottom right are 'SAVE' and 'CANCEL' buttons.

FIGURE 6.7: User Successfully Updates Asset - Update Asset - Step 1 - 5

The screenshot shows the 'Asset Details' view for the asset 'Testing Asset'. The asset details are: Name: Testing Asset, Asset ID: rKj0sKEilHgQV3a3M3fN, Type: Data, Reviewed: Testing, Description: Testing. To the right, the 'Owner Details' section shows: Owner: Aphral Griffin, Department: Engineering, Contact: aphral.griffin+1@gmail.com. Below this is the 'Asset Risks' section, which is currently empty. At the bottom left are 'DELETE', 'EDIT', 'REVIEW', and 'ADD RISK' buttons.

FIGURE 6.8: Cancel Button Clicked - Update Asset - 2a

The screenshot shows the 'Update Asset' interface. At the top, there's a header bar with 'RISKABLE' on the left and 'SIGN OUT' on the right. Below the header is a sidebar with 'Home', 'Asset', and 'Controls' options. The main area has a title 'Update Asset' and a sub-section 'Update Asset: Testing Editing Asset'. It contains four input fields: 'Asset Name' (Testing Editing Asset), 'Asset Type' (Data), 'Asset Description' (Edit Testing), and 'Asset Department' (Sales). Below these fields is an error message: 'Error: Problems Updating Asset'. At the bottom are 'SAVE' and 'CANCEL' buttons.

FIGURE 6.9: Error Updating Asset - Update Asset - 3a

This screenshot shows the 'Delete Asset' process at step 2. The main content area displays asset details: Name (Laptops), Asset ID (EY2rQY4nlbSBgcoaNl), Type (Equipment), Reviewed (24/04/2022), and Description (Company Laptops used by employees daily. Brought in and out of the office weekly.). To the right is a 'Owner Details' section with information for Aph Griffin (Owner, Department: Engineering, Contact: aphral.griffin@gmail.com) and an 'Asset Risks' table. The table has two rows: Theft (High, Medium) and Damage (Medium, Medium). A modal dialog at the top center asks 'localhost:3000 says Are you sure you want to delete this Asset?' with 'Cancel' and 'OK' buttons. At the bottom are 'DELETE', 'EDIT', 'REVIEW', and 'ADD RISK' buttons.

FIGURE 6.10: Warning Message When Deleting Asset - Delete Asset - Step 2

This screenshot shows the 'Delete Asset' process at step 3. The main content area is identical to Figure 6.10, displaying asset details and owner information. The modal dialog now shows 'localhost:3000 says Success Deleting Asset' with an 'OK' button. The 'Asset Risks' table below is empty, indicating 'No rows'. At the bottom are 'DELETE', 'EDIT', 'REVIEW', and 'ADD RISK' buttons.

FIGURE 6.11: Successfully Deleted Asset - Delete Asset - Step 3 - 5

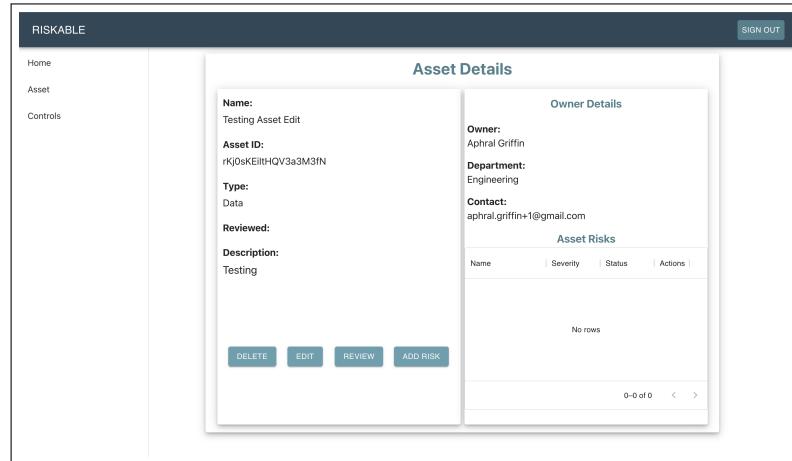


FIGURE 6.12: User Presses Cancel - Delete Asset - 2a

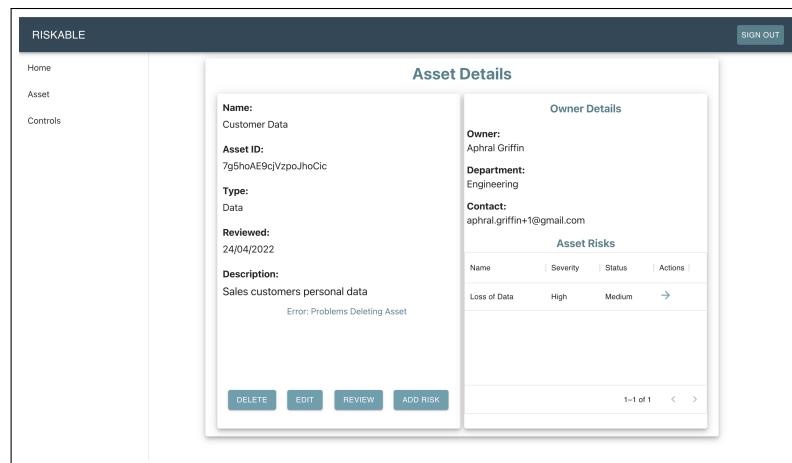


FIGURE 6.13: Error Deleting Asset - Delete Asset - 3a

6.1.3 Use Case 3 - Maintain Risk

Title: Add Risk		
Main Success Scenario	Step	Description
E: Employee S: System	1	E: Fills in Risk Details & Selects Relevant Controls
	2	E: Clicks Add Risk Button
	3	S: Adds Risk To Database
	4	S: Displays Success Message
	5	S: Brings Employee to Asset Detail Page
Extensions	2a	Cancel Button Clicked S: Brings Employee to Asset Details Page
	3a	Error Adding Risk S: Display Error Message

TABLE 6.5: Use Case 3 - Maintain Risk - Add Risk

Title: Update Risk		
Main Success Scenario	Step	Description
E: Employee S: System	1	S: Populates Fields with Risk Details
	2	E: Fills in Risk Details
	3	E: Clicks Update Risk Button
	4	S: Updates Risk In Database
	5	S: Displays Success Message
	6	S: Brings Employee to Risk Details Page
Extensions	2a	Cancel Button Clicked S: Brings Employee to Risk Details Page
	3a	Error Updating Risk S: Display Error Message

TABLE 6.6: Use Case 3 - Maintain Risk - Update Risk

Title: Delete Risk		
Main Success Scenario	Step	Description
E: Employee S: System	1	E: Clicks Delete Button
	2	S: Displays Warning asking Employee are They Sure
	3	E: Clicks Confirm
	4	S: Deletes Risk From Database
	5	S: Displays Success Message
	6	S: Brings Employee to Asset Details Page
Extensions	2a	Cancel Button Clicked S: Brings Employee to Asset Details Page
	4a	Error Updating Risk S: Display Error Message

TABLE 6.7: Use Case 3 - Maintain Risk - Delete Risk

The screenshot shows the 'Add Risk' interface. At the top, a message says 'localhost:3000 says Success Adding Risk'. The main form has fields for Risk Name ('Testing Risk'), Risk Status ('Medium'), Risk Severity ('High'), Risk Description ('Testing'), and Tags To Label Risk. To the right, a sidebar titled 'Control Suggestions' lists various risk control measures. At the bottom are 'SAVE' and 'CANCEL' buttons.

FIGURE 6.14: User Successfully Adds Risk - Add Risk - Step 1 - 4

The screenshot shows the 'Asset Details' page for an asset named 'Customer Data'. It displays owner details (Owner: Aphra Griffin, Department: Engineering, Contact: aphra.griffin+1@gmail.com) and a table of asset risks (Loss of Data, Severity: High, Status: Medium). At the bottom are buttons for 'DELETE', 'EDIT', 'REVIEW', and 'ADD RISK'.

FIGURE 6.15: Cancel Button Clicked - Add Risk - 2a

The screenshot shows the 'Add Risk' form. The 'Risk Name*' field contains 'Testing Risk Add'. The 'Risk Status' dropdown is set to 'Medium'. The 'Risk Severity' dropdown is set to 'High'. The 'Risk Description*' field is empty. Below the form, an error message says 'Error: Problem Adding Risk'. At the bottom are 'SAVE' and 'CANCEL' buttons.

FIGURE 6.16: Error Adding Risk - Add Risk - 3a

The screenshot shows the 'Update Risk: Loss of Data' form. The 'Risk Name' field contains 'Loss of Data'. The 'Risk Status' dropdown is set to 'Medium'. The 'Risk Severity' dropdown is set to 'High'. The 'Risk Description' field contains 'Loss of all customer data would have long lasting effects on the company'. A success message at the top says 'localhost:3000 says Success Updating Risk'. At the bottom are 'SAVE' and 'CANCEL' buttons.

FIGURE 6.17: User Successfully Updates Risk - Update Risk - Step 1 - 5

The screenshot shows the 'Risk Details' page for the risk 'Loss of Data'. The left panel displays risk details: Name (Loss of Data), Risk ID (jkGLB0Xzkri3dQDSPml), Status (Medium), Severity (High), Last Reviewed (24/04/2022), and Description (Loss of all customer data would have long lasting effects on the company). The right panel shows 'Owner Details' (Owner: Aphral Griffin, Department: Engineering, Contact: aphral.griffin+1@gmail.com) and 'Risk Controls' (Backups: Daily Backups of Servers). At the bottom are 'DELETE RISK', 'EDIT RISK', and 'REVIEW RISK' buttons.

FIGURE 6.18: Cancel Button Clicked - Update Risk - 2a

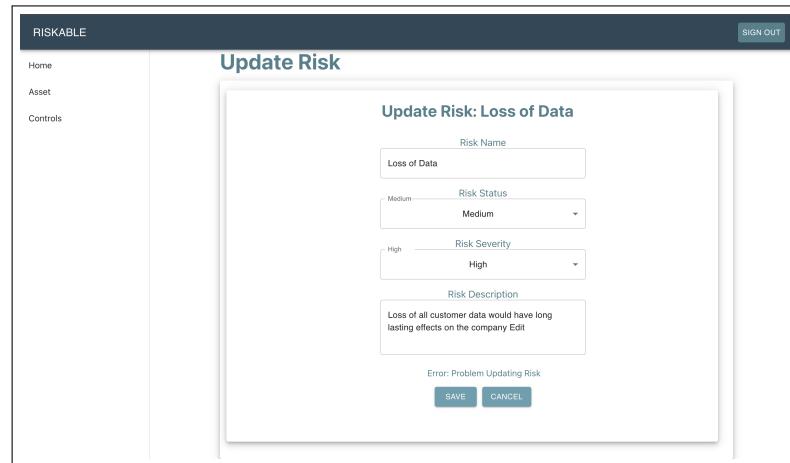


FIGURE 6.19: Error Updating Risk - Update Risk - 3a

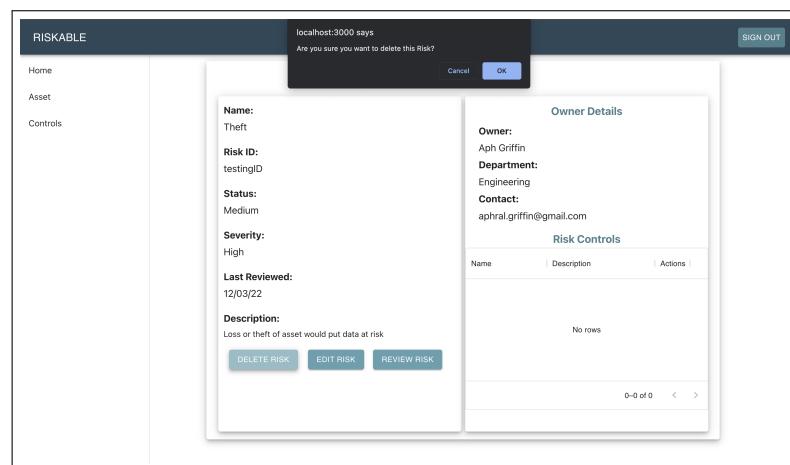


FIGURE 6.20: Warning Message When Deleting Risk - Delete Risk - Step 2

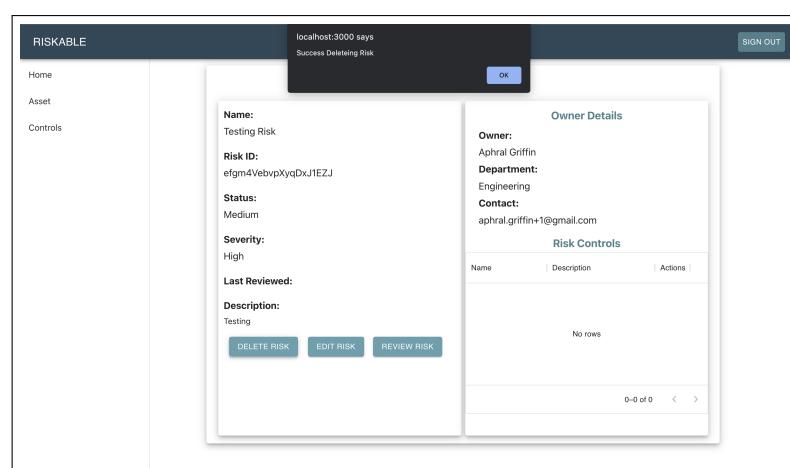


FIGURE 6.21: Successfully Deleted Risk - Delete Risk - Step 3 - 5

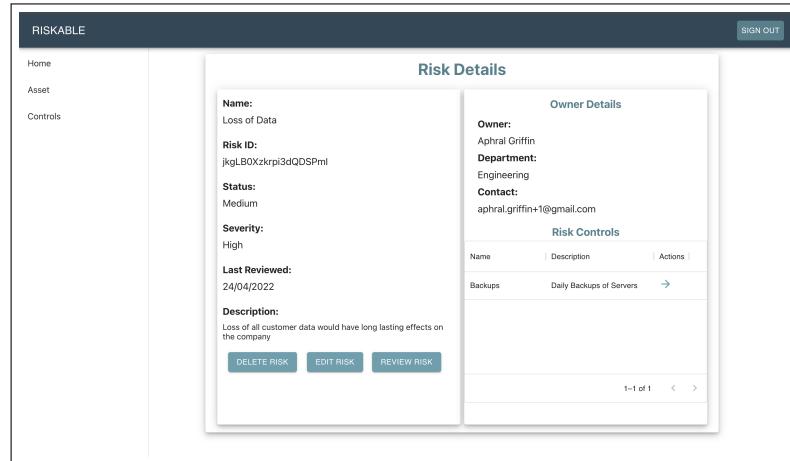


FIGURE 6.22: User Presses Cancel - Delete Risk - 2a

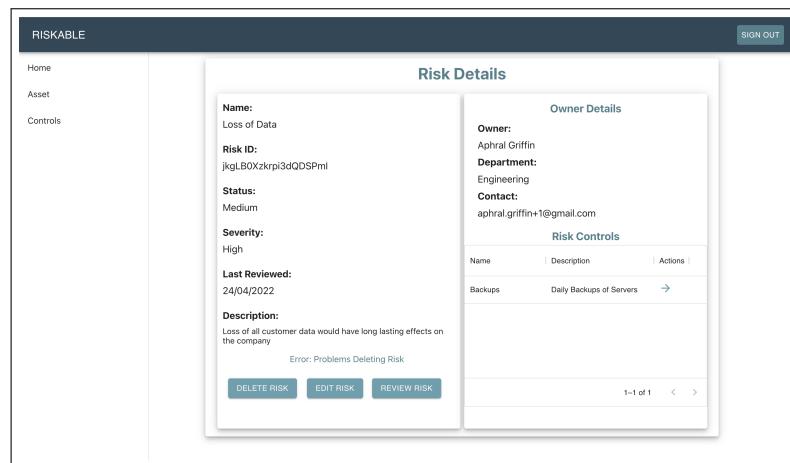


FIGURE 6.23: Error Deleting Risk - Delete Risk - 3a

6.1.4 Use Case 4 - Maintain Control

Title: Add Control		
Main Success Scenario	Step	Description
E: Employee S: System	1	E: Fills in Control Details
	2	E: Clicks Add Control Button
	3	S: Adds Control To Database
	4	S: Displays Success Message
	5	S: Brings Employee to Control List Page
Extensions		Cancel Button Clicked
	2a	S: Brings Employee to Control List Page
	3a	Error Adding Control
		S: Display Error Message

TABLE 6.8: Use Case 4 - Maintain Control - Add Control

Title: Update Control		
Main Success Scenario	Step	Description
E: Employee S: System	1	S: Populates Fields with Control Details
	2	E: Fills in Control Details
	3	E: Clicks Update Control Button
	4	S: Updates Control In Database
	5	S: Displays Success Message
	6	S: Brings Employee to Control Details Page
Extensions		Cancel Button Clicked
	2a	S: Brings Employee to Control Details Page
	3a	Error Updating Control
		S: Display Error Message

TABLE 6.9: Use Case 4 - Maintain Control - Update Control

Title: Delete Control		
Main Success Scenario	Step	Description
E: Employee S: System	1	E: Clicks Delete Button
	2	S: Displays Warning asking Employee are They Sure
	3	E: Clicks Confirm
	4	S: Deletes Control From Database
	5	S: Displays Success Message
	6	S: Brings Employee to Controls List Page
Extensions	Cancel Button Clicked	
	2a	S: Brings Employee to Control Details Page
	Error Updating Control	
	4a	S: Display Error Message

TABLE 6.10: Use Case 4 - Maintain Control - Delete Control

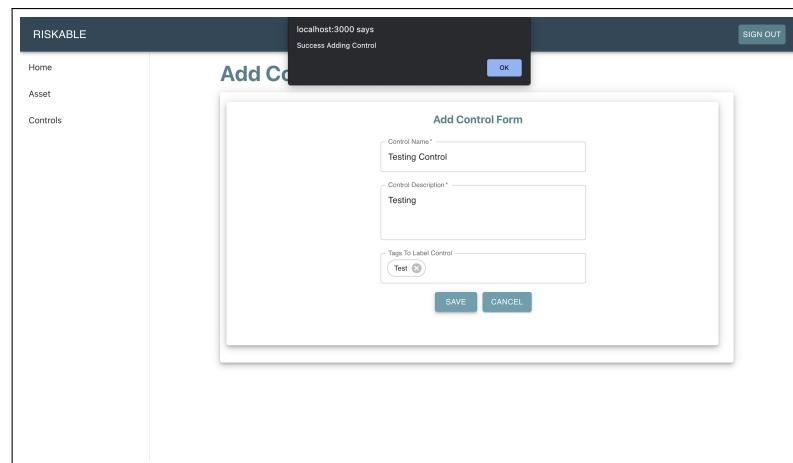


FIGURE 6.24: User Successfully Adds Control - Add Control - Step 1 - 4

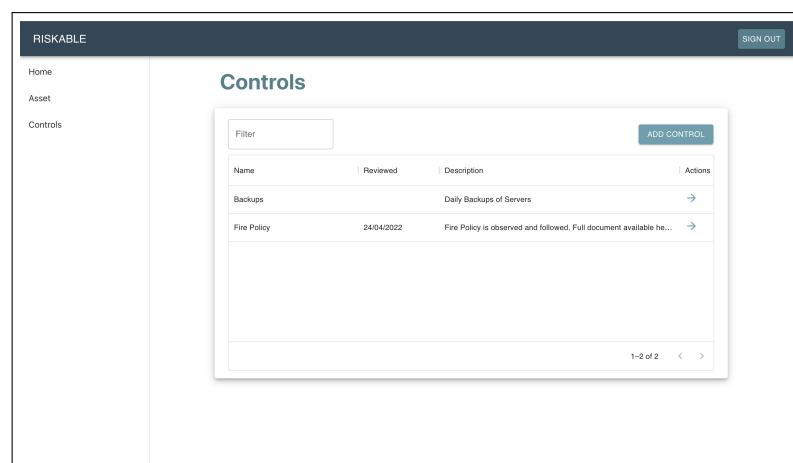


FIGURE 6.25: Cancel Button Clicked - Add Control - 2a

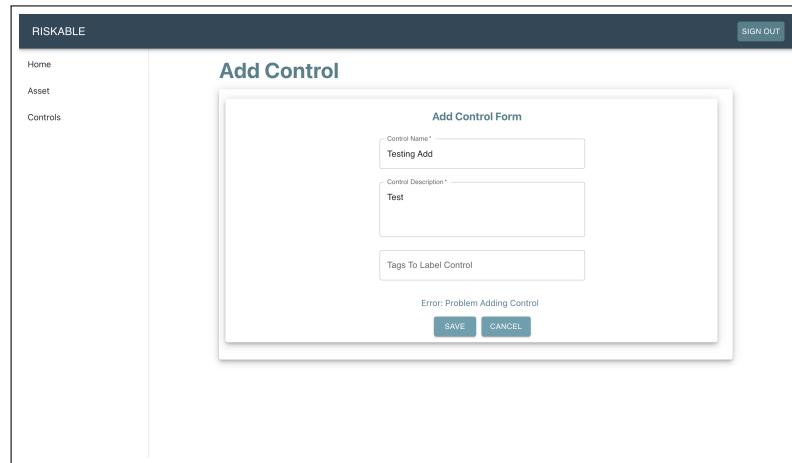


FIGURE 6.26: Error Adding Control - Add Control - 3a

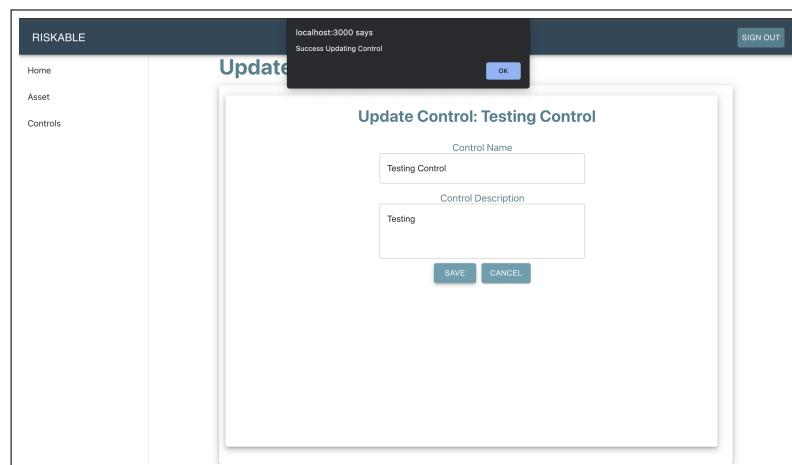


FIGURE 6.27: User Successfully Updates Control - Update Control - Step 1 - 5

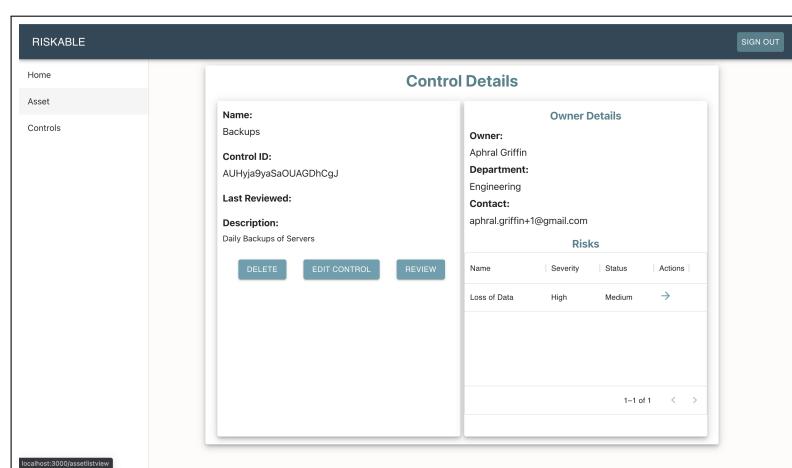


FIGURE 6.28: Cancel Button Clicked - Update Control - 2a

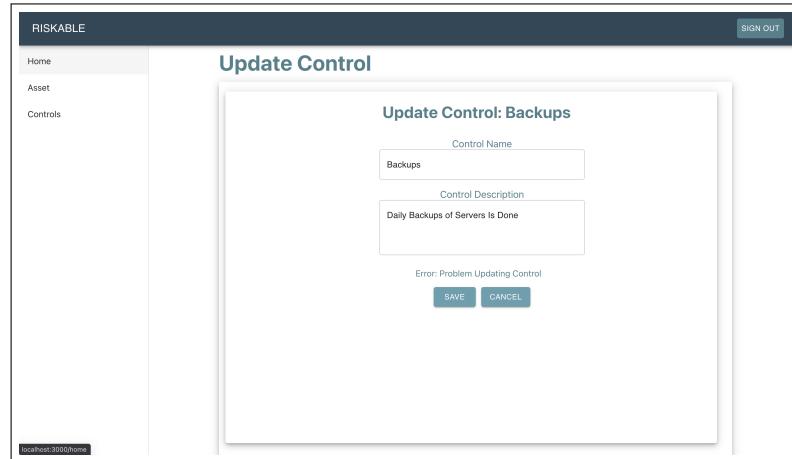


FIGURE 6.29: Error Updating Control - Update Control - 3a

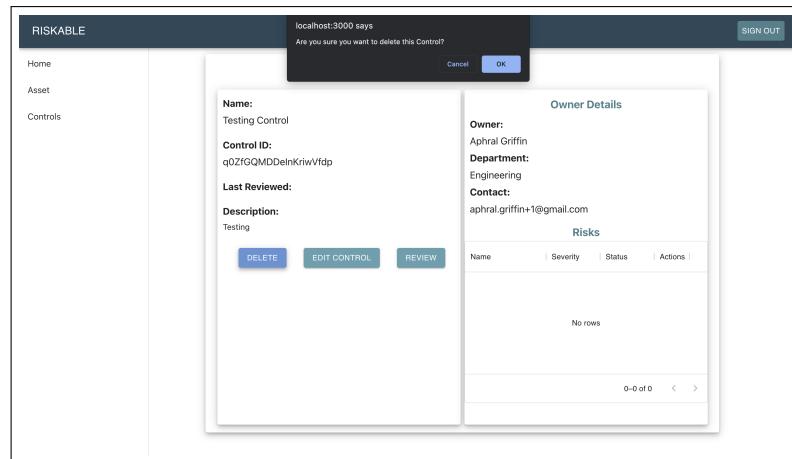


FIGURE 6.30: Warning Message When Deleting Control - Delete Control - Step 2

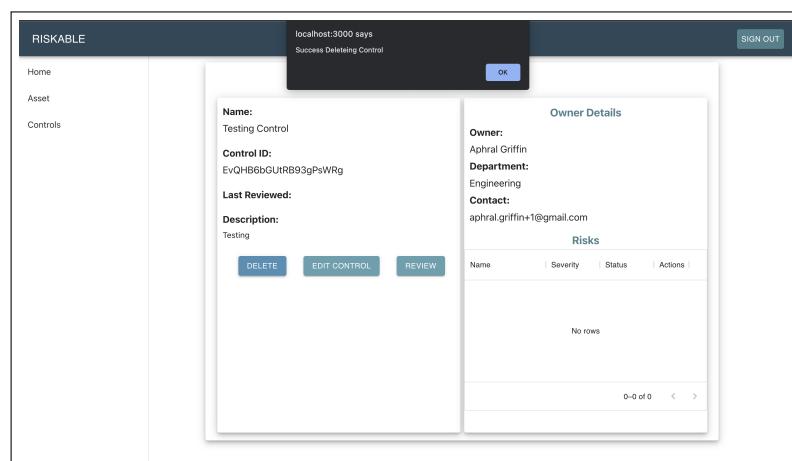


FIGURE 6.31: Successfully Deleted Control - Delete Control - Step 3 - 5

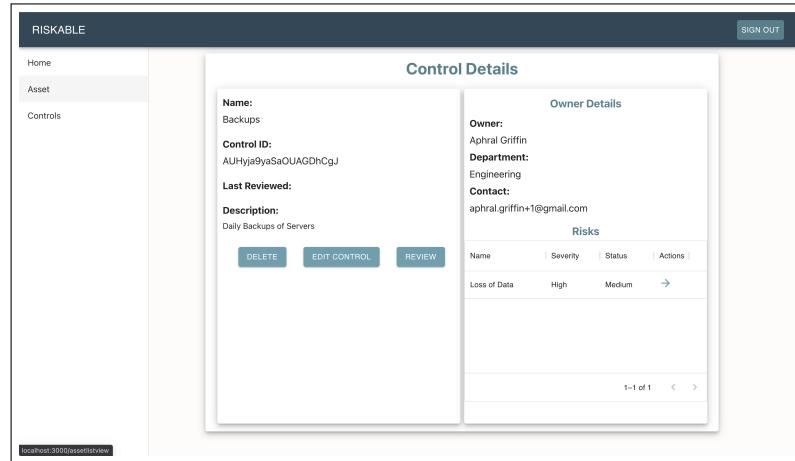


FIGURE 6.32: User Presses Cancel - Delete Control - 2a

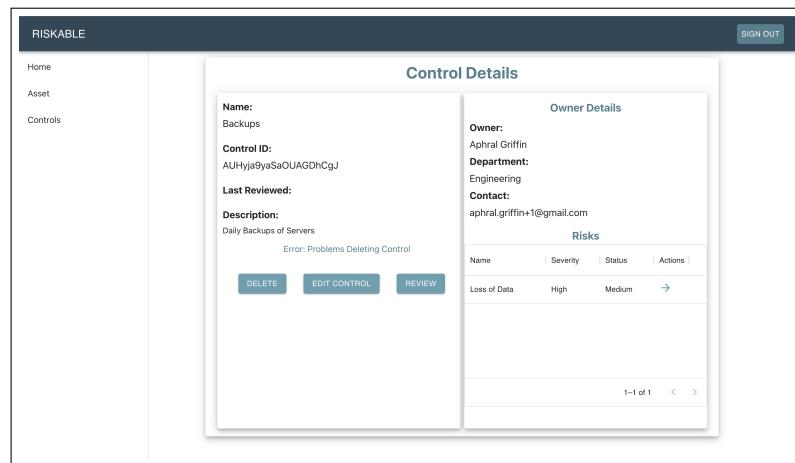


FIGURE 6.33: Error Deleting Control - Delete Control - 3a

6.1.5 Use Case 5 - View Assets, Risks & Controls

Title: View Assets		
Main Success Scenario	Step	Description
E: Employee S: System	1	E: Clicks on Assets in Nav Bar.
	2	S: Brings Employee to Asset List Page

TABLE 6.11: Use Case 5 - View Asset

Title: View Risks		
Main Success Scenario	Step	Description
E: Employee S: System	1	E: Clicks on Assets on Asset List Page.
	2	S: Brings Employee to Asset Details Page
	3	E: Clicks on Risks on Asset Details Page.
	4	S: Brings Employee to Risk Details Page

TABLE 6.12: Use Case 5 - View Risk

Title: View Controls		
Main Success Scenario	Step	Description
E: Employee S: System	1	E: Clicks on Controls in Nav Bar.
	2	S: Brings Employee to Control List Page

TABLE 6.13: Use Case 5 - View Control

The screenshot shows the RISKABLE application's user interface. At the top, there is a dark header bar with the word "RISKABLE" on the left and a "SIGN OUT" button on the right. Below the header, there is a vertical navigation menu on the left side with three items: "Home", "Asset", and "Controls". The main content area is titled "Assets". It features a search bar labeled "Filter" at the top. Below the search bar is a table with columns: "Name", "Type", "Department", and "Actions". The table contains six rows of data:

Name	Type	Department	Actions
Customer Data	Data	Sales	→
Laptops	Equipment	Engineering	→
Building	Building	Management	→
Warehouse	Building	Warehouse	→
Sales Leads	Data	Sales	→

At the bottom of the table, there is a small text "1-6 of 7" followed by navigation arrows. On the far right of the main content area, there is a "ADD ASSET" button.

FIGURE 6.34: User Views Assets - Step 1 & 2

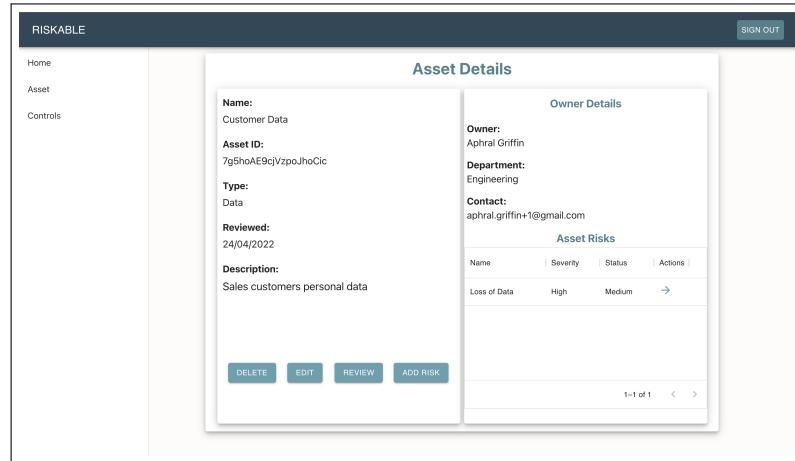


FIGURE 6.35: User Views Risks on Asset Page - Step 1 - 2

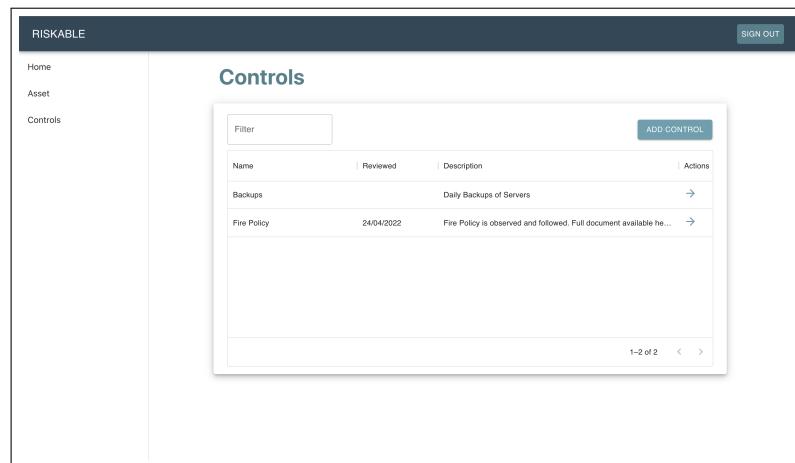


FIGURE 6.36: User Views Controls - Step 1 & 2

6.1.6 Use Case 8 - Tagging

Title: Tagging		
Main Success Scenario	Step	Description
E: Employee S: System	1	E: Enters Details of Asset or Risk
	2	E: Adds Tags To Asset or Risk
	3	S: Filters Risks or Controls by Tags
	4	S: Displays Relevant Data
	5	E: Selects Relevant Options and Adds to Asset or Risk
Extensions	3a	No Relevant Risks or Controls S: Display Empty List

TABLE 6.14: Use Case 8 - Tagging

RISKABLE

Add Asset

Add Asset Form

Asset Name*	Testing Asset
Asset Associated Department*	Sales
Asset Type*	Data
Asset Description*	Testing
Tags To Label Asset	Testing

Risk Suggestions

Name	Tags
Inspection	Building,Fire
Backup	Data
Training	Data,Email

1-3 of 3 < >

SAVE **CANCEL**

FIGURE 6.37: User Enters Details with No Tags - Step 1

RISKABLE

Add Asset

Add Asset Form

Asset Name*	Testing Adding Asset
Asset Associated Department*	Sales
Asset Type*	Data
Asset Description*	Testing
Tags To Label Asset	Data

Risk Suggestions

Name	Tags
Backup	Data
Training	Data,Email

1 row selected 1-2 of 2 < >

SAVE **CANCEL**

FIGURE 6.38: User Can View Filtered Data - Step 2 - 4

RISKABLE

Add Asset

Add Asset Form

Asset Name*	Testing Asset
Asset Associated Department*	Sales
Asset Type*	Data
Asset Description*	Testing
Tags To Label Asset	Testing

Risk Suggestions

Name	Tags
------	------

No rows

0-0 of 0 < >

SAVE **CANCEL**

FIGURE 6.39: No Relevant Data - Step 4a

6.2 Portability Testing - Non-Functional Requirements

To test the portability of the web application, the web application was accessed on multiple devices, operating systems and through different browsers. The tests are detailed below along with screenshots of the application on each device, operating system and browser.

6.2.1 Different Operating Systems

The application was accessed on both a PC running a Windows operating system and a laptop running a Mac Operating System to test that the web application would work on different operating systems.

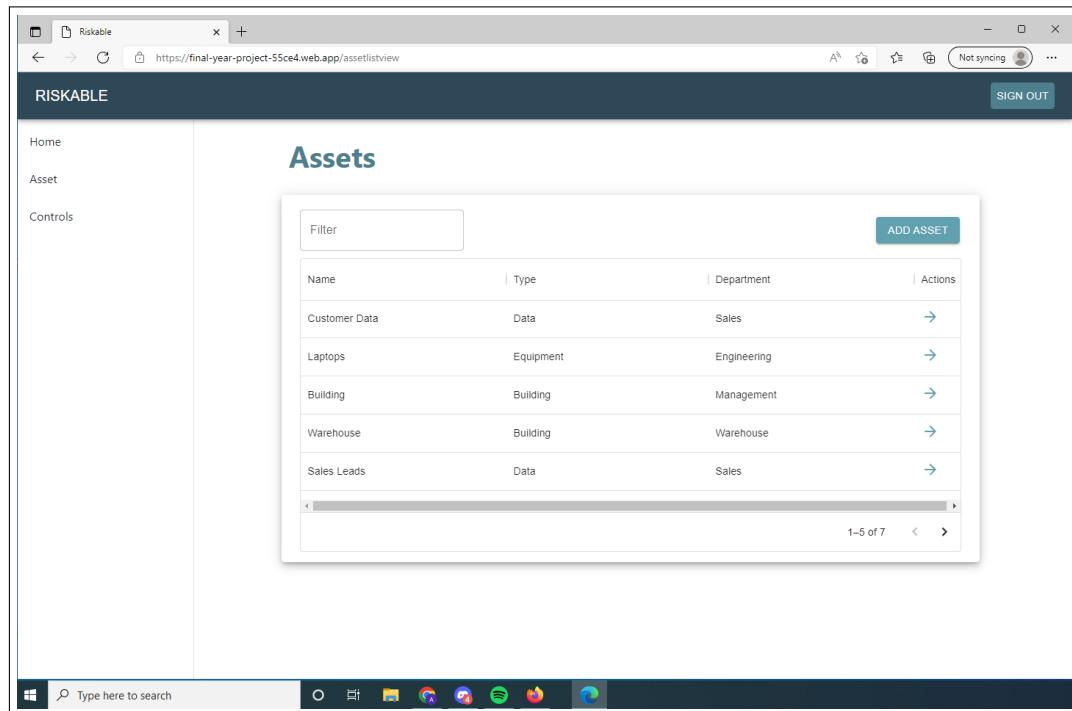


FIGURE 6.40: Web Application Working on Windows

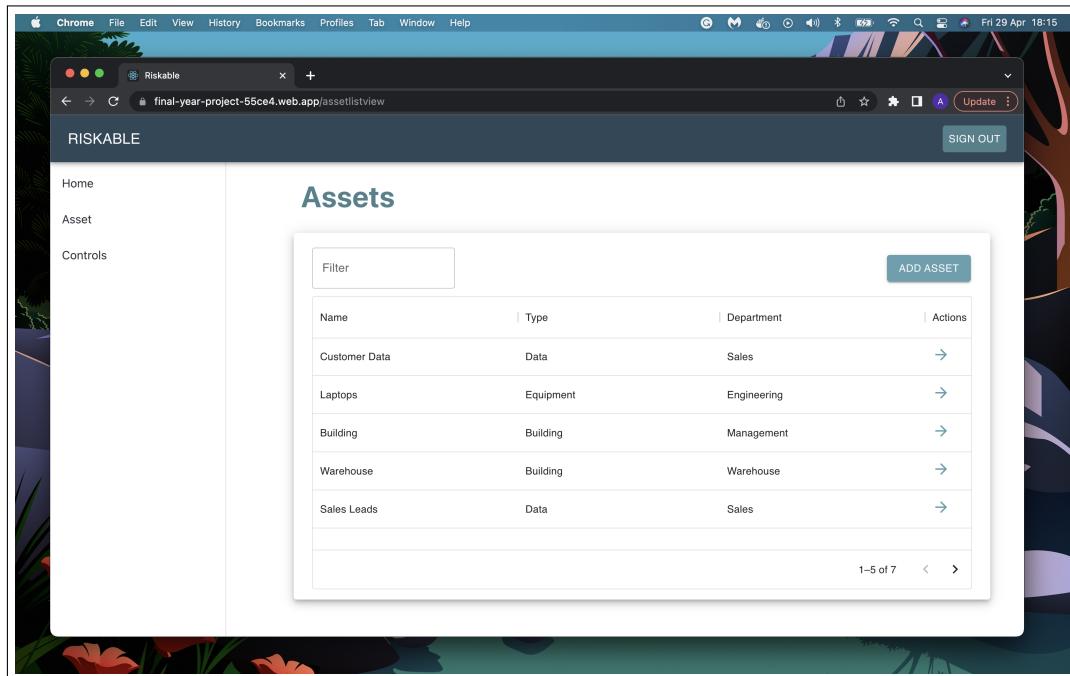


FIGURE 6.41: Web Application Working on Mac

6.2.2 Different Devices

To test that the application would work on different devices, the web application was accessed on a laptop, iPhone and Android phone.

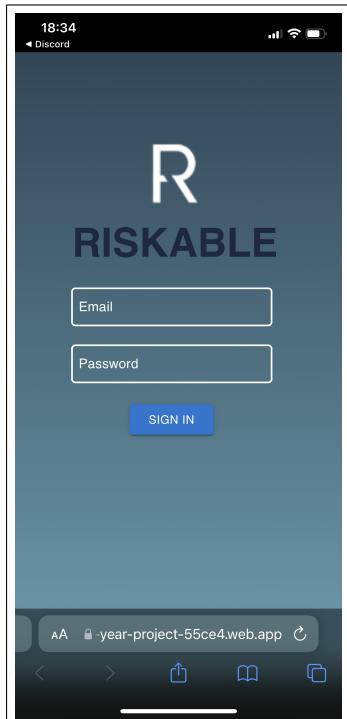


FIGURE 6.42: Web Application Working on iPhone

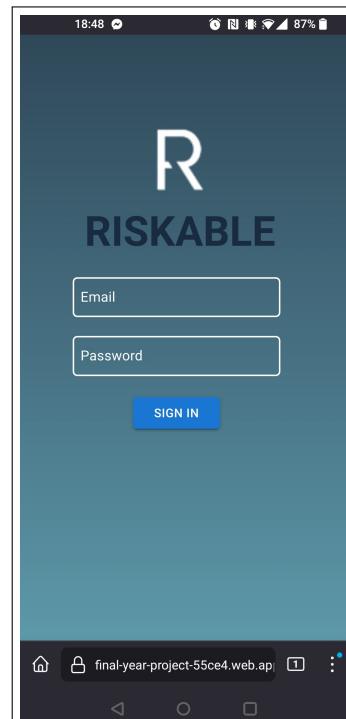


FIGURE 6.43: Web Application Working on Android

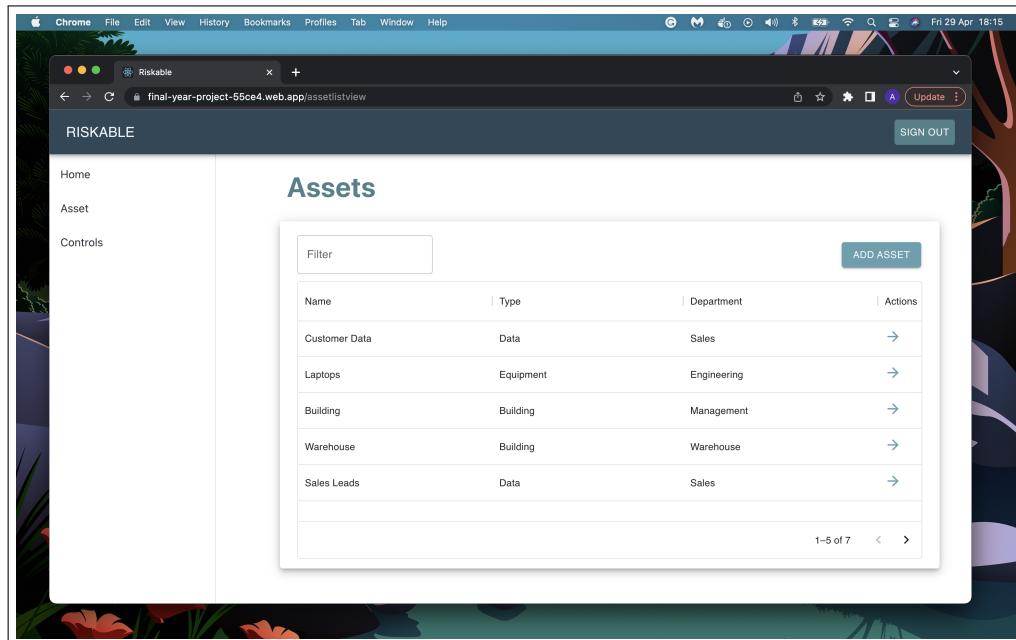


FIGURE 6.44: Web Application Working on Laptop

6.2.3 Different Browsers

To test that the application would work on different browsers, the web application was accessed on Chrome, Microsoft Edge, Firefox and Safari.

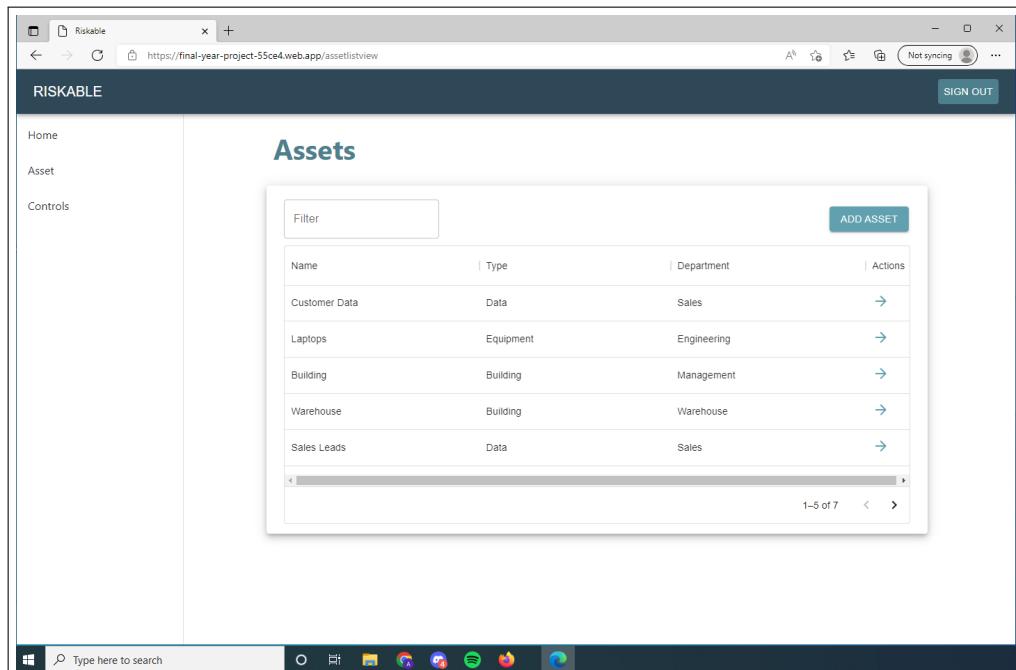


FIGURE 6.45: Web Application Working on Microsoft Edge

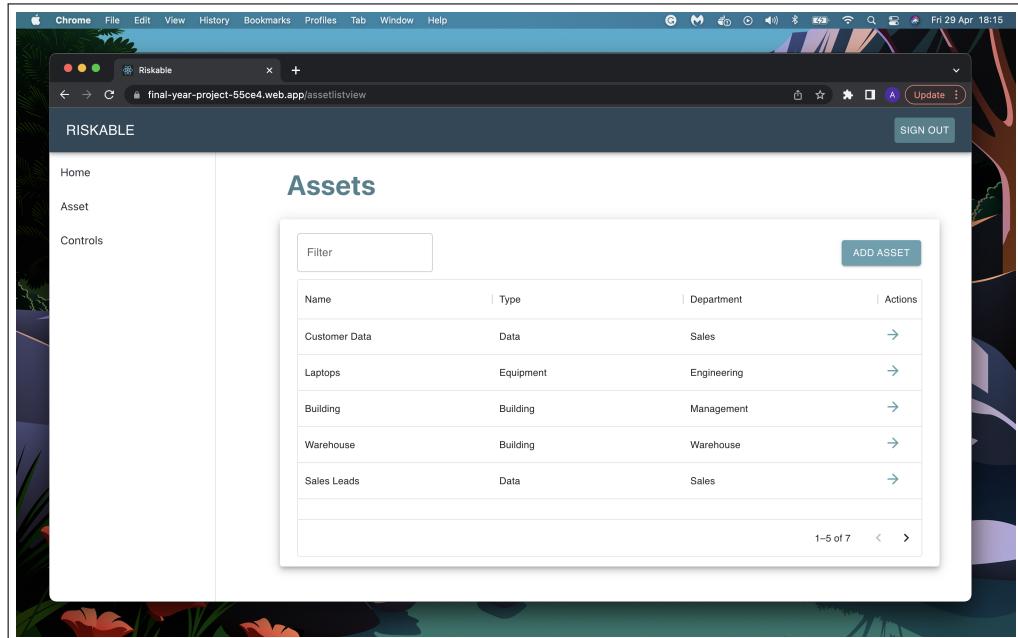


FIGURE 6.46: Web Application Working on Chrome

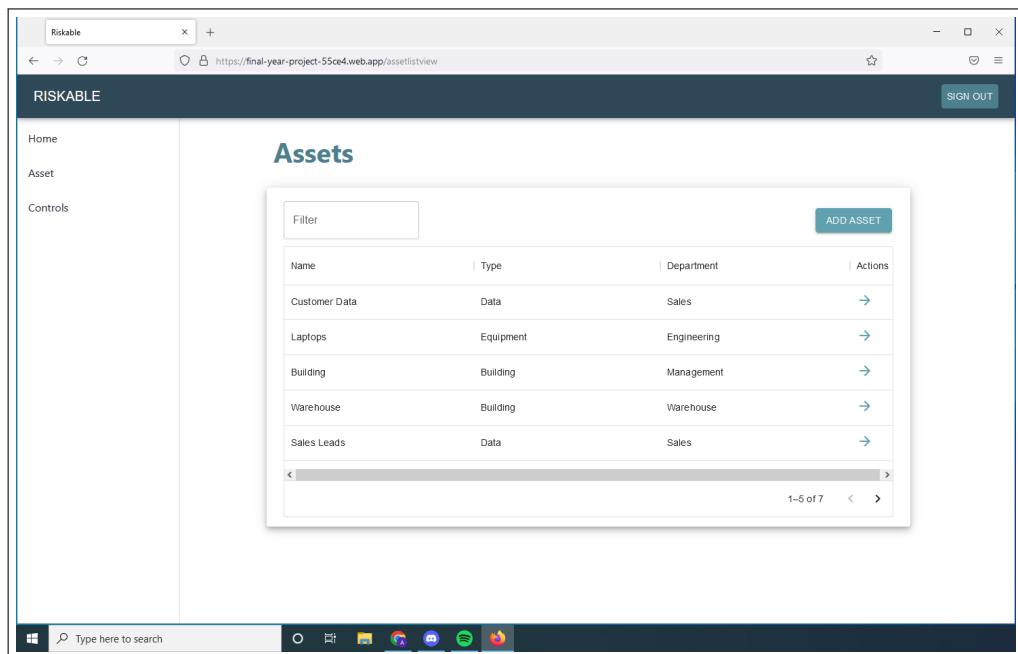


FIGURE 6.47: Web Application Working on Firefox

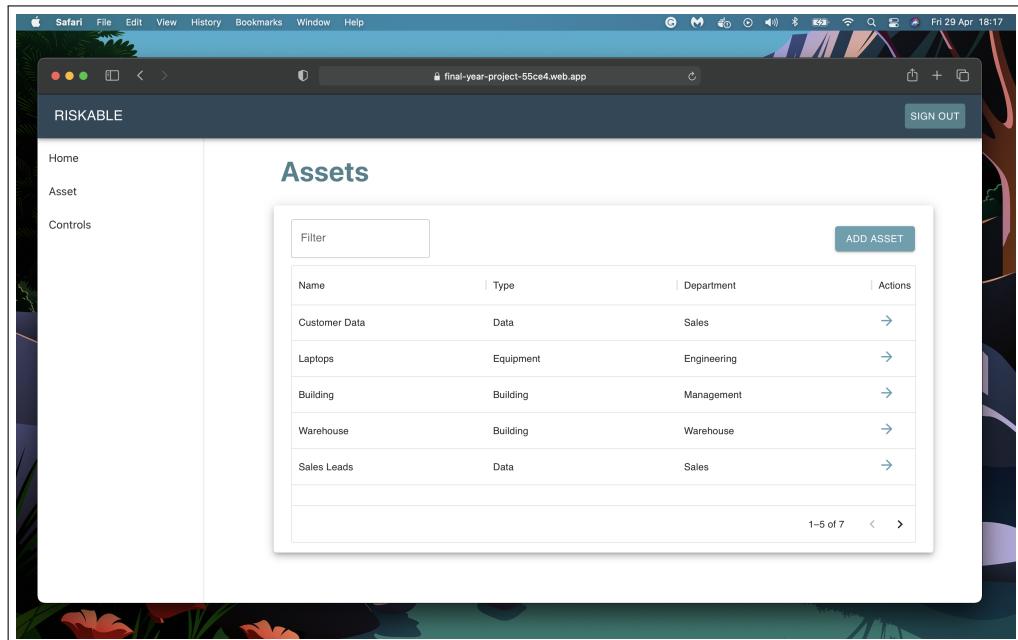


FIGURE 6.48: Web Application Working on Safari

6.3 Results

The results of the tests specified in the two previous sections are displayed in the tables below.

6.3.1 Use Case Testing Results

There were 14 use case tests which were carried out on the application. All of these tests were passed by the application.

Use Case Name	Description	Pass/Fail	Notes
Login	User can Login	Pass	
Add Asset	User can Add Asset	Pass	
Update Asset	User can Edit Asset	Pass	
Delete Asset	User can Delete Asset	Pass	
Add Risk	User Can Add Risk	Pass	
Update Risk	User can Update Risk	Pass	
Delete Risk	User can Delete Risk	Pass	
Add Control	User can Add Control	Pass	
Update Control	User can Update Control	Pass	
Delete Control	User can Delete Control	Pass	
View Asset	User can View Asset	Pass	
View Risk	User can View Risk	Pass	
View Control	User can View Control	Pass	
Tagging	User can View Suggested Data	Pass	
Percentage of Tests Passed:	100%		

TABLE 6.15: Use Case Testing Results

6.3.2 Portability Testing Results

There were multiple portability tests carried out on the application. All of the tests were passed but a note has been made on the mobile tests, this is because though the application was functional on mobile it wasn't very user friendly.

Name	Description	Pass/Fail	Notes
Mac Operating System	Web application accessed on Mac OS	Pass	
Windows Operating System	Web application accessed on Windows	Pass	
Android	Web application accessed on Android	Pass	UI needs to be fixed to improve user experience
iPhone	Web application accessed on iPhone	Pass	UI needs to be fixed to improve user experience
Laptop	Web application accessed on Laptop	Pass	
Chrome	Web application accessed using Chrome	Pass	
Firefox	Web application accessed using Firefox	Pass	
Safari	Web application accessed using Safari	Pass	
Microsoft Edge	Web application accessed using Microsoft Edge	Pass	
Percentage of Tests Passed:	100%		

TABLE 6.16: Portability Testing Results

6.3.3 Industry Product Tests

If this web application was adopted by industry there are many more tests that would need to be carried out. These tests would include but are not limited to the following.

- **Scalability Testing :** The system should be tested to see if it can handle heavy use and scale as it needs to. This would hopefully not be an issue as Firebase automatically scales.
- **Security Testing :** The system should be tested to see if it is vulnerable to attacks. This would be done by checking to see if attacks such as the current OWASP top ten[37] are possible and protected against along with hiring an external company to perform a penetration test.
- **Reliability Testing :** The system should be tested to see how reliable it is under different conditions. This would be done by monitoring the system under different conditions for a period of time to check for failures.
- **Usability Testing :** The system should be tested to see if it is user friendly. This would be done by getting a group of people to try perform a set of tests like adding an asset, and seeing how the users find the system are there many errors etc.
- **Availability Testing :** The system should be tested to see if it has issues under different conditions. The system should be monitored to see how it performs under different conditions and how that affects the user's experience.

Chapter 7

Discussion and Conclusions

7.1 Solution Review

The problem this project is trying to solve is to provide a way for companies to better manage company assets and their associated risks. During the research phase of this project, a web application that would allow users to log in and manage their assets, risks and controls was researched and planned. The application would also make suggestions to the user about risks and controls in order to help users who may not have much knowledge on managing assets and the standards that might apply to them.

During the implementation phase of this project, a web application was developed using React, JavaScript and Firebase. In the previous chapter, this web application was tested based on the use cases described in the research phase of this project. These use cases described the functionality the application would need in order to try to solve the problem. As can be seen in the previous chapter results the web application passed all of these tests.

The application developed allows users to log in, and manage their assets, risks and controls. The application also provides suggestions to the user in order to help them get set up on the system quickly and protect their assets.

7.2 Project Review

Though the application developed for this project does address the problem the project was trying to solve. If this project was to be done again there are a few things that would be done differently.

The first one is using Typescript at the start of the project, though this language was used on work placement, building a project from scratch with TypeScript was very different to the way it was used on work placement. The lack of tutorials on building a project with TypeScript and Firebase version 9 made it very difficult to get Firebase working. If this was to be done again more time would be put into checking the available tutorials are up to date with both technologies before choosing to use the language.

The second is spending more time learning about Firebase, the changes between version 8 and version 9 of Firebase were extreme. Though they make the experience of using Firebase better, the changes meant that most of the information available about Firebase was no longer up to date. This meant a lot of time was spent searching for answers to simple problems and though the Firebase documentation is up to date it isn't easy for new users to understand as is a basic outline of the product's functionality.

The third and final is using MUI to its fullest potential. MUI provides a way to make themes which then can be used throughout the application easily to change the style of its components. This would have removed some of the repetitive CSS and other styling code used throughout the project. MUI themes weren't used in this project as there was a lack of experience with it and there wasn't time to learn and implement the theme.

Throughout this project there were multiple difficulties encountered, most of these difficulties were to do with the changes to Firebase and using TypeScript at the beginning of the project. When a problem was discovered it was researched online to try to find a solution, if a solution was not found it was brought up to the project supervisor during weekly meetings to see if they had any suggestions. A solution was found to all the problems encountered throughout the project though some took longer than others to solve.

Through this project, I have improved my knowledge of JavaScript, React and Firebase along with gaining more experience using Go. Time and project management were huge elements of this project, this project involved prioritising work, distributing my time between multiple modules along with this project and planning sprints. All of the experience and skills gained in this project will be very useful throughout my future career.

7.3 Future Work

If there had been more time available to spend on this project there are a number of features that would have been implemented. These features include but are not limited to the features described below.

- **Input Validation**

Though input validation is simple to implement it is a time-consuming process and after discussing it with the project supervisor, it was decided to instead focus on implementing the main features of the application instead of validation. Input validation would improve the usability of the app and decrease user error allowing for a better overall user experience.

- **Google Docs Suggestions**

During the research phase of this project, a feature was planned which would allow the user to get suggestions from an AI about policies that would apply to an asset, risk or control. This was going to be done using Google Docs API and spaCy. When the AI was replaced with tagging the plan was to still implement the Google Docs policy suggestions but due to time constraints, this feature was not implemented. In the future, if this feature were to be implemented the user would tag a Google Doc in the web application. Then when adding a new asset, control or risk the application would display a list of policy documents with those tags that may apply to the asset, risk or control.

- **Forgot Password Page**

At the moment there is no way for a user to reset their password unless they contact the admin who has access to the Firebase project. There was a plan to have a forgot password page which would allow the user to reset their password. Due to time constraints, this page was not implemented but if there was more time it would be done using React, JavaScript and using Firebase Authentication `sendPasswordResetEmail` and `updatePassword` functions. The user would enter their email and Firebase Authentication would send an email with a link for them to reset their password.

- **Add User Page**

To add a user to the system, an admin who has access to the Firebase project has to add a user and set a password for that user within the Firebase project Authentication section. If there was more time to work on this project an add user page would be implemented. This page would only be available to users with the Admin role, it would contain a form which the admin can fill in the new users, username and password. The admin would then send these login details to the user.

- **AI**

Originally an AI was planned for this project but after discussion with my supervisor, a decision was made to drop the AI as it was overcomplicating the system and tagging would provide the same functionality. If there was more time and data from the users, an AI could be used to automatically tag assets, risks, controls and documents instead of having the user tag them. This could be done using spaCy which was researched in the research phase of this project.

7.4 Conclusion

This project set out to try to provide a way for companies to better manage company assets and their associated risks. The project also aimed to try and help smaller companies that may not have much knowledge about managing assets and risks.

The research phase of this project demonstrated a need to have an application that was available to smaller companies. There are applications available to manage assets but all of the applications found were paid and geared towards bigger companies.

The implementation phase of this project has developed a web application that addresses this problem using React and JavaScript. The web application developed passes all the use cases that were created during the research phase of this project. The web application allows users to easily manage their assets, risks and controls.

The application also has a tagging system which allows the user to tag assets, risks and controls. This then allows the system to make suggestions to the user about which risks or control may apply to an asset or risk as they are adding that asset or risk. This helps the user better protect their assets and get set up on the system faster.

The web application developed solves the problem this project was trying to address. It provides a way for any company to manage their company assets and associated risks. While also having a way to help smaller companies that might have a harder time getting started.

Bibliography

- [1] TypeScript, “Type script.” [Online]. Available: <https://www.typescriptlang.org/>
- [2] React, “React.” [Online]. Available: <https://reactjs.org/>
- [3] M. UI, “Mui.” [Online]. Available: <https://mui.com/>
- [4] “Firebase documentation,” <https://firebase.google.com/docs>.
- [5] spaCy, “spacy.” [Online]. Available: <https://spacy.io/usage/spacy-101>
- [6] Google, “Google doc api.” [Online]. Available: <https://developers.google.com/docs/api>
- [7] M. Martin, “Difference between website and web application (web app),” <https://www.guru99.com/difference-web-application-website.html>, 10 2021.
- [8] O. of Global Operations and U. o. K. Security, “Security management and planning why is security awareness important?” <https://gos.ku.edu/security%20mangement>.
- [9] E. J. W. Boers, W. A. Kosters, P. van der Putten, M. Poel, and J. N. Kok, “Artificial intelligence-artificial intelligence: Definition, trends, techniques and cases-joost n artificial intelligence: Definition, trends, techniques, and cases.”
- [10] S. A. Grishaeva and V. I. Borzov, “Information security risk management.” Institute of Electrical and Electronics Engineers Inc., 9 2020, pp. 96–98.
- [11] ISO, “Standards,” <https://www.iso.org/standards.html>.
- [12] S. V. Aleksandrova, V. A. Vasiliev, and M. N. Aleksandrov, “Problems of implementing information security management systems.” Institute of Electrical and Electronics Engineers Inc., 9 2020, pp. 78–81.
- [13] J. Hampton, *Fundamentals of Enterprise Risk Management: How Top Companies Assess Risk, Manage Exposure, and Seize Opportunity*, 2nd ed. AMACOM, 2014, 2014.
- [14] P. Hopkin, *Fundamentals of Risk Management Understanding, Evaluating and Implementing Effective Risk Management*, 5th ed. Kogan Page Publishers, 2018, 2018.

- [15] “Rims conference,” <https://www.rims.org/rims2021/rims-2021-save-the-date>.
- [16] “The 13 best risk ’&’ security conferences of 2021,” <https://www.resolver.com/blog/best-risk-security-conferences/>.
- [17] E. Burke, “What’s going on with the hse cyberattack?” <https://www.siliconrepublic.com/enterprise/hse-cyberattack-explainer-conti-ransomware>, 5 2021.
- [18] “Kaspersky lab (2020),” <https://www.kaspersky.ru>.
- [19] Angraini, Megawati, and L. Haris, “Risk assessment on information asset an academic application using iso 27001.” Institute of Electrical and Electronics Engineers Inc., 3 2019.
- [20] M. J. Wheeldon, J. Hayes, and B. C. MICE, “Developing a whole company culture of asset management through organisational structure, an asset management framework and a risk based approach for asset intervention,” 2013.
- [21] S. Fenz, J. Heurix, T. Neubauer, and F. Pechstein, “Current challenges in information security risk management,” *Information Management and Computer Security*, vol. 22, pp. 410–430, 11 2014.
- [22] R. Montesino and S. Fenz, “Information security automation: How far can we go?” 08 2011, pp. 280–285.
- [23] S. Fenz, A. Ekelhart, and E. Weippl, “Fortification of it security by automatic security advisory processing,” 01 2008, pp. 575–582.
- [24] E. A. Fenz S and N. T, “Business process-based resource importance determination,” 2009, pp. 113–127.
- [25] H.-S. Rhee, Y. Ryu, and C. Kim, “Unrealistic optimism on information security management,” *Computers & Security*, vol. 31, p. 221–232, 03 2012.
- [26] “What is the isms.online platform?” www.isms.online/software-features/the-isms-online-platform/.
- [27] “Vigilant software.” [Online]. Available: <https://www.vigilantsoftware.co.uk/topic/cyber-comply-tool>
- [28] “Firebase documentation,” <https://firebase.google.com/docs/hosting>.
- [29] “Firebase documentation,” <https://firebase.google.com/docs/firestore>.
- [30] “Firebase documentation,” <https://firebase.google.com/docs/functions>.
- [31] “Firebase documentation,” <https://firebase.google.com/docs/auth>.

- [32] GitHub, “Github docs.” [Online]. Available: <https://docs.github.com/en/get-started/quickstart/hello-world>
- [33] V. S. Code, “Visual studio code.” [Online]. Available: <https://code.visualstudio.com/docs>
- [34] Go, “Go.” [Online]. Available: <https://go.dev/>
- [35] “Trello.” [Online]. Available: <https://trello.com/>
- [36] C. Gawin, “Why the new firebase web v9 modular sdk is a game-changer.” dev community.” [Online]. Available: <https://dev.to/chroline/why-the-new-firebase-web-v9-modular-sdk-is-a-game-changer-nph>
- [37] OWASP, “Owasp top ten.” [Online]. Available: <https://owasp.org/www-project-top-ten/>