ICT 3103/3203: Secure Software Development
Trimester 1 AY22/23

Report I

Secure Software Requirement Analysis and Design

Team Information	Team Members	Student ID
Grp 38 (P4)	CHONG FU MIN	2002039
	PANG XUE MING	2000989
	TAN YU HUI	2001632
	TIFFANY HO	2002170
	YEO SWAN YEW	2000981
	KERMAN CHAN	2001189

Glossary

This section contains the table for the definitions, acronyms and abbreviations used within this document.

Table 1: Glossary Table

Term	Description
AAA	Authentication, Authorisation, Accounting. A framework to understand security issues relating to accessibility
API	Application programming interface
CIA	Confidentiality, Integrity, Availability. A model to describe the goals of cybersecurity
DDOS	Distributed Denial of Service attack. An attack that uses multiple computers or machines to flood a targeted resource.
FR	This refers to the Functional Requirements
FSR	This refers to the Functional Security Requirements
Manage	All CRUD is represented by the word “Manage” for the functional requirements
MAF	This refers to the Multi-Factor Authentication
NFSR	This refers to the Non Functional Security Requirements
NR	This refers to the Non-functional Requirements
PDPA	This refers to the Personal Data Protection Act
SFR	This refers to the Secure Functional Requirements
SQL	This refers to Structured Query Language
WAF	This refers to Web Application Firewall
2FA	This refers to 2 Factor Authentication

1. Overview of Application

1.1 Project Background

Our application is an E-commerce, Pastel De Luna that sells mooncakes to customers. Upon visiting our site, the customer will be brought to the home page which they can browse the mooncake Pastel De Luna has to offer. Customers are free to browse but will be required to sign up for an account before purchasing mooncakes. We provide customization of the customer’s profile (including their allergy, if any) and various payment methods for the customers. Customers are allowed to choose their delivery method as well. Occasionally, promotion codes will be available for customers to apply discounts upon payment. Overall, Pastel De Luna should be a smooth and secure site for customers.

1.2 Stakeholder and Intended Users

Our stakeholders are the members of this team, the professors of this module, our customers who browse Pastel De Luna’s web application. It also includes editors who will provide catalogue update

information for the administrators to provide access control, handles and keeps the database up to date.

Table 2: Stakeholders and Intended Users

Stakeholders	Description
Users	Intended Users, anyone who visits Pastel De Luna
Customers	Any user who registered an account with Pastel De Luna
Administrator	An employee who provides access control, handles and keeps databases up to date, ensures security measures for databases, also creates accounts for employees
Editor	An employee who maintains Pastel De Luna's website with the latest catalogue
Developers	The team members who design and develop Pastel De Luna's website
Professors	Accessors of the module

1.3 Benefits of Application

Table 3 explains the benefits of having Pastel De Luna's web application.

Table 3: Benefits of Application

Benefits	Explanation
Impact	Our website mainly benefits mooncake lovers
Creativity	Use of promotion codes for customers to enjoy mooncake at a discounted price
Usefulness	Our website is more efficient and effective compared to setting up a physical shop, there is no queuing needed for browsing and payment compared to physical shops

2. Business Requirements

Section 2 describes the business requirements that are required of Pastel De Luna. These requirements are described based on functional requirements and non-functional requirements of the application.

2.1 Functional Requirements

Table 4 displays the functional requirements of the application. These shall be fulfilled to ensure the web application can provide the basic functionality of a Business to Consumer (B2C) e-commerce system.

Table 4: Functional requirements of Application

S/N	Requirement Description
FR01	Users/Customers/Editor shall be able to browse the website.
FR02	Users/Customers/Editor shall be able to search for mooncake using the search bar.
FR03	Users shall be able to register for an account.
FR04	Customers/Editor/Administrators shall be able to perform login and logout.
FR05	Customers shall be able to manage their own profile details.
FR06	Customers shall be able to manage the mooncakes from the cart.
FR07	Customers shall be able to make payment.
FR08	Customers shall receive a confirmation email when purchasing mooncake.
FR09	Customers shall be able to view payment history.
FR10	Customers shall be able to view their order statuses: confirmed, processing, delivered.
FR11	Administrators shall be able to create accounts for the Editor through the database.
FR12	Administrators shall be able to delete accounts for the Editor through the database.
FR13	Administrators shall be able to manage requests for the products.
FR14	Editors shall be able to send requests to manage the products.

2.2 Non-functional Requirements

Table 5 describes the non-functional requirements which are the Software Quality Attributes (SQA) that the application should possess.

Table 5: Non-Functional requirements of Application

S/N	Requirement Description	SQA
NR01	The website shall not take more than 5 seconds to load web contents.	Performance
NR02	Additional code backup to provide redundancy.	Reliability
NR03	The website shall be able to retrieve and process data and information from the database within 5 seconds.	
NR04	Code will be written in a way that favours the implementation of new functions or features for both corrective and enhancement maintenance.	Maintainability

NR05	The website shall ensure high availability of 99.9%.	Availability
NR06	Database server shall ensure high availability of 99.9%.	
NR07	The website shall be able to run on multiple devices such as phones and tablets.	Compatibility
NR08	The website shall be able to run on multiple browsers like Chrome, Firefox, etc.	
NR09	The website shall be able to prevent unauthorised access to any authorised pages.	Accessibility
NR10	The website shall be responsive to users.	Usability
NR11	The website shall be organised and pleasing to the eyes.	

3. Security Requirements

Section 3 explains the security measures that need to be implemented for the Pastel De Luna web application. The security requirements are categorised into secure functional requirements, functional security requirements, and non-functional security requirements as shown in section 3.1, 3.2, and 3.3 respectively.

3.1 Secure Functional Requirements

Table 6 depicts what should not happen for each of the functional requirements in the e-commerce application.

Table 6: Secure Functional Requirements of Application

S/N	Requirement Description	Mapped To
SFR01	Customers shall not be able to login without a password	FR03, FR04, FR05
SFR02	Users shall not be able to manage other customer profile details	FR05
SFR03	Customers shall not be able to manage other customer profile details	FR03, FR05, FR09
SFR04	Customers shall not be able to view other customers carts	FR06, FR09
SFR05	Customers shall not be able to make a purchase without payment details	FR05, FR07, FR09, FR10
SFR06	Customer shall not be able to update the product information	FR14
SFR07	Customer shall not be access the webpage in HTTP internet protocol	FR01
SFR08	Customers shall not be able to search other information besides mooncake details from the search bar	FR02

SFR09	Customer shall not be able to receive other customer confirmation email for the payment of mooncake	FR08
SFR10	Administrator shall not be able to directly manage the product without receiving request from editors	FR13
SFR11	Editors shall not be able to update the database directly for the product informations	FR14
SFR12	User/Customers shall not be able to create editor account	FR11, FR12

3.2 Functional Security Requirements

Table 7 depicts the functional security requirements for the application based on the CIA model and AAA framework to ensure the goals of cybersecurity are achieved and issues relating to accessibility are minimised in the context of an e-commerce application.

Table 7: Functional Security Requirements of Application

S/N	Requirement Description	Category
FSR01	Data transferred on the website must be done via HTTPs for more security and prevent domain spoofing.	Confidentiality
FSR02	Data stored in the database should be encrypted to prevent readability	Confidentiality
FSR03	The website shall not display explicit technical information (eg: error codes, server configurations) that are useful for an attacker	Confidentiality
FSR04	The website shall store sensitive data in sessions instead of cookies to prevent data manipulation	Integrity
FSR05	The website shall validate all user inputs from the search bar to reduce the possibility of injection attacks.	Integrity
FSR06	The website shall have an additional backup system implemented for data/system.	Availability
FSR07	The website shall have an additional database.	Availability
FSR08	The website shall ensure implement a proper network content delivery to mitigate DDoS	Availability
FSR09	The website shall have a log management implemented for web server and database server.	Accountability
FSR10	There should be MFA/2FA authentication before successful account registration	Authentication
FSR11	There should be MFA/2FA authentication before payment	Authentication

FSR12	The website shall ensure a Secure Token authentication for pages that require customer to login (eg: view profile, view cart, payment, etc)	Authentication
FSR13	Registration Page should perform input validation for all fields to ensure compliance with registration requirements and prevent injection attacks	Authentication
FSR14	The website shall have a proper access control implemented for both end-user and back-end administrator	Authorisation

3.2 Non-Functional Security Requirements

Table 8 depicts the security requirements relating to the architecture of Pastel De Luna's web application.

Table 8: Non-Functional Security Requirements of Application

S/N	Requirement Description	Relevance	Category
NFSR01	The website shall have a downtime of not more than 4 hours	To ensure Pastel De Luna is still able to continue business and users are able to continue access in event of DDOS attack	Reliability
NFSR02	The website shall have a 99.999% uptime as high availability	To ensure Paste De Luna is always ready to serve its customers	Availability
NFSR03	The website shall ensure secure transactions via verified and trusted API.	This is important because this would involve customer's money and Pastel De Luna's business	Security
NFSR04	The website shall have a session time-out of 20 min of inactive activity	This is important as it will help to prevent any possible way of session-based attack	Security
NFSR05	The website shall close all unnecessary ports to reduce attack surface.	This is important as it prevent hacker to access the VM through other port.(e.g SSH)	Security
NFSR06	The website shall have log monitoring of every days	To trace the activity done by each of the users so that suspicious events can be detected and mitigate early	Traceability
NFSR07	The encryption algorithm used to encrypt sensitive data should be fast	To ensure smoothness of user experience	Performance
NFSR08	The encryption algorithm used to decrypt sensitive data should be fast	To ensure smoothness of user experience	Performance

NFSR09	The website shall collect and store data of customers in a way that is compliant to the Personal Data Protection Act (PDPA).	To ensure customers information will not be disclose to the public and to adhere to regulation	Privacy
NFSR10	The rendering of the browser after 2FA/MFA authentication shall not take more than 5 sec	To ensure that the authentication process is efficient and reliable for the public.	Reliability

4. Use Case and Misuse Case diagram

Section 4 describes Pastel De Luna's possible attacks and misuse cases (section 4.2 - 4.5) that are derived from the functional requirements (section 4.1 and 4.1.1) of the business.

4.1 Use Case Diagram

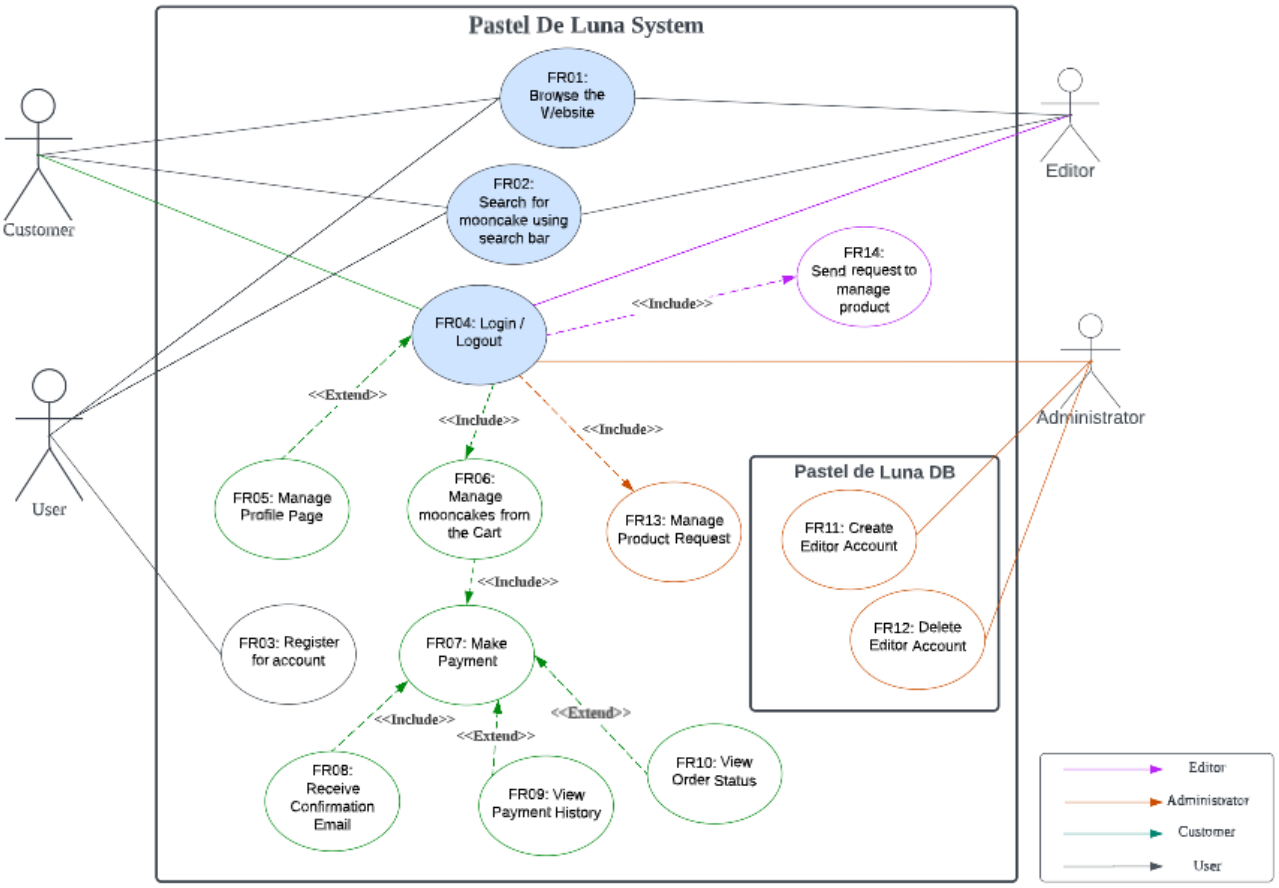

Figure 1: Use Case Diagram

4.1.1. Use Case Description

Section 4.1.1 illustrates an example of use case descriptions from customer, user, administrator and editor. For the full list of use case description, refer to Appendix A: Use Case Description.

Table 9: Customer use case - Make payment

Use Case ID - Name	**UC7 - Make payment**
Actor	Customer
Description	The use case allows customer to make payment online for the mooncakes added to the cart.

Table 10: User use case - Register for account

Use Case ID - Name	**UC3 - Register for account**
Actor	User
Description	The use case allows users to register for a customer account so that they are able to login to continue their orders.

Table 11: Administrator use case - Manage Product Request

Use Case ID - Name	**UC13 - Manage Product Request**
Actor	Administrator
Description	The use case allows administrator to receive a product change request given by the editor so that they can make the necessary changes on the database.

Table 12: Editor use case - Send Request to manage the Product

Use Case ID - Name	**UC14 - Send Request to manage the Product**
Actor	Editor
Description	The use case allows editor to send a product change request to the administrator so that the product catalogue content can change accordingly.

4.2 Customer Misuse Case Diagram

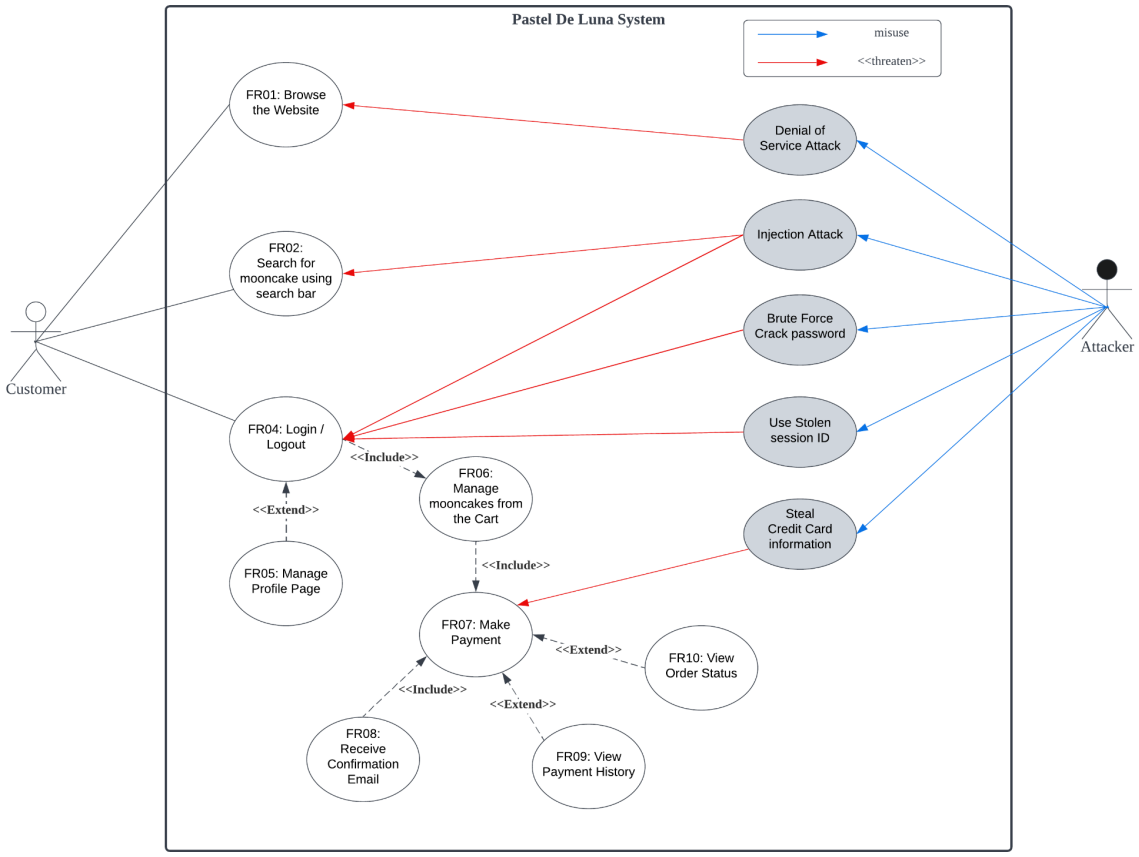

Figure 2: Customer Misuse Case Diagram

4.2.1. Customer Misuse Case Description

Table 13: Customer MUC1

Misuse Case ID - Name	**MUC1 - Denial of Service Attack**
Actor	Customer, Attacker
Description	Attackers can perform denial-of-Service (DoS) attacks by flooding our web server which can potentially shut down our web application, making browsing inaccessible to customers.

Table 14: Customer MUC2

Misuse Case ID - Name	**MUC2 - Injection Attack**
Actor	Customer, Attacker
Description	Injection attacks like SQL injection could threaten the register and login page, or the search bar where customer inputs are needed.

Table 15: Customer MUC3

Misuse Case ID - Name	**MUC3 - Brute Force Crack Password Attack**
Actor	Customer, Attacker
Description	Attackers can employ trial and error and other applications to crack passwords, login credentials, and encryption keys in the login feature of the web application.

Table 16: Customer MUC4

Misuse Case ID - Name	**MUC4 -Use Stolen Session ID**
Actor	Customer, Attacker
Description	Attackers trick a customer into logging in to a fake website that is similar to the Pastel De Luna website, once the customer enters their login details, the attacker will receive it and try to login to the real website with their details.

Table 17: Customer MUC5

Misuse Case ID - Name	**MUC5 -Steal Credit Card Information**
Actor	Customer, Attacker
Description	In the event of insecure communications (eg: http) The attacker manages to perform Man in the middle (MITM), and non-encrypted data can be seen in plaintext. Attacker can obtain the credit card information.

4.3 User Misuse Case Diagram

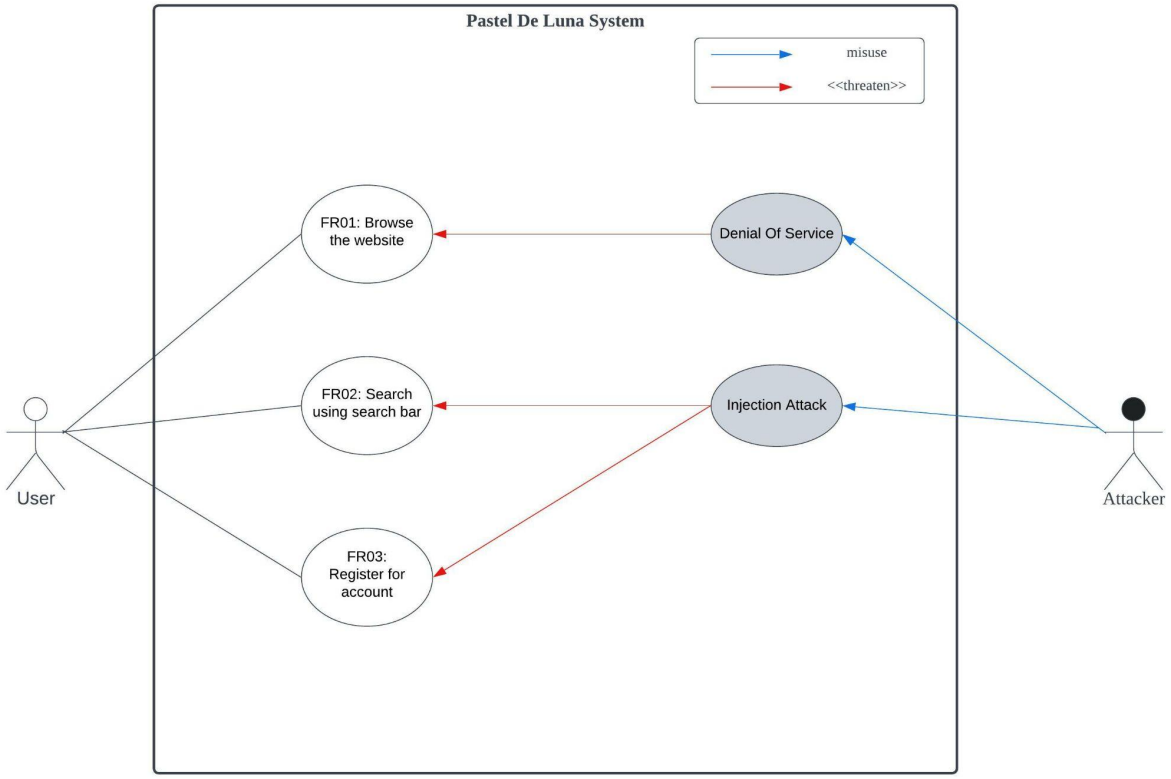

Figure 3: User Misuse Case Diagram

4.3.1. User Misuse Case Description

Table 18: User MUC6

Misuse Case ID - Name	**MUC6 - Denial of Service Attack**
Actor	User, Attacker
Description	Attackers can perform denial-of-Service (DoS) attacks by flooding our web server which can potentially shut down our web application, making browsing inaccessible to customers.

Table 19: User MUC7

Misuse Case ID - Name	**MUC7 - Injection attack**
Actor	User, Attacker
Description	Injection attacks like SQL injection could threaten the register and login page, or the search bar where customer inputs are needed.

4.4 Administrator Misuse Case Diagram

The administrator is able to create and delete the Editor's account from the database which is external from what our website is able to provide. The administrator is only able to access the website to manage the request asked from the Editor.

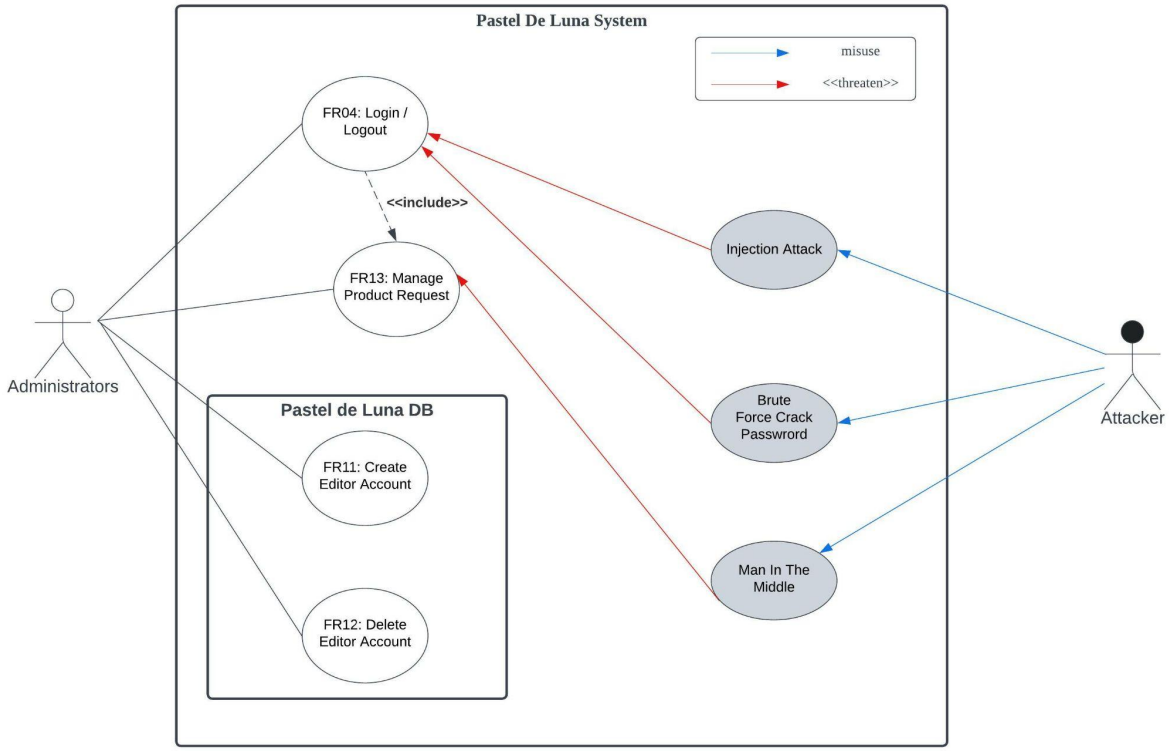

Figure 4: Administrator Misuse Case Diagram

4.4.1. Administrator Misuse Case Description

Table 20: Administrator MUC8

Misuse Case ID - Name	**MUC8 - Injection Attack (SQL Injection)**
Actor	Administrator, Attacker
Description	Injection attack could be done by submitting an SQL like query input to any vulnerable input fields such as the login fields.

Table 21: Administrator MUC9

Misuse Case ID - Name	**MUC9 - Brute Force Crack Password Attack**
Actor	Administrator, Attacker
Description	Attackers can employ trial and error and other applications to crack passwords, login credentials, and encryption keys in the login feature of our web application.

Table 22: Administrator MUC10

Misuse Case ID - Name	MUC10 - Man In the Middle
Actor	Administrator, Attacker
Description	An execution that intercepts the admin user traffic through the attacker's network before the request reaches its intended destination to the database.

4.5 Editor

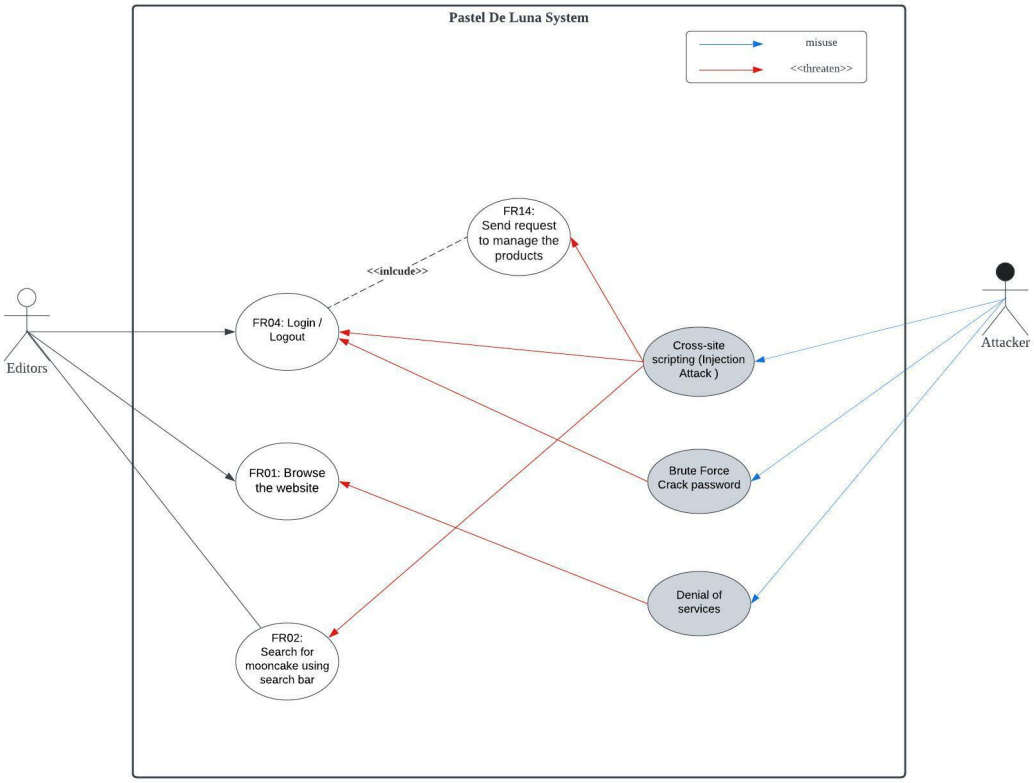

Figure 5: Editor Misuse Case Diagram

4.5.1. Editor Misuse Case Descriptions

Table 23: Editor MUC11

Misuse Case ID - Name	MUC11 - Injection Attack (XSS)
Actor	Editor, Attacker
Description	Injection attacks such as SQL injection and malicious script can be used by attacker to compromise the registration and login page field validation, as well as the search bar where user inputs are required on our web application.

Table 24: Editor MUC12

Misuse Case ID - Name	MUC12 - Brute Force Crack Password Attack
Actor	Editor, Attacker
Description	Attackers can employ trial and error and other applications to crack passwords, login credentials, and encryption keys in the login feature of our web application.

Table 25: Editor MUC13

Misuse Case ID - Name	MUC13 - Denial of Service Attack
Actor	Editor, Attacker
Description	Attackers can perform denial-of-Service (DoS) attacks by flooding our back-end web server which can potentially shut down our web application, making it inaccessible to editors.

5. Potential Risk of Application

Table 26: Software Criticality Classification

Risk Level	Impact Description
High	Any outage in the identified risk results in immediate cessation of primary functions of our system, equivalent to immediate and critical impact to our system functionality and customer satisfaction.
Medium	Any outage in the identified risk results in cessation over time or an immediate reduction of a primary function of the systems, equivalent to minor impact to our system functionality, and customer satisfaction.
Low	A sustained outage in the identified risk results in little to no impact on a primary function of our system.

Table 27: Identification of Potential Risk

S/N	Potential Risk	Risk Description	Critical Level	Rank of criticality (1 being most critical)
1	Denial of service (Web/database Platform downtime)	Adversary can flood huge amounts of traffic to our web server causing disruption or shutting down our web server, rendering it inaccessible to customers.	**High**	**1**

2	Sensitive Data Leakage	Adversaries can perform unauthorised access and transmission of customer sensitive data stored on our web or system database to their end-host or destination, leading to sensitive data leak.	**High**	**2**
3	Injection Attack - Cross-Site Scripting (XSS) & SQL Injection	Injection of malicious script from an adversary into our front-end website's input fields such as login, register and search field to exploit possible vulnerability in the input field validation.	**High**	**3**
4	Session-hijacking	Exploiting our website session control to gain access to a customer user's session token results in unauthorised access to sensitive customer data.	**High**	**4**
5	Secure authentication failure	Failure of an authentication feature like 2FA/MFA will make it considerably easier for an adversary to go around the authentication protection on our website's login and payment features.	**High**	**5**
6	Insecure Communication	Sensitive data that is being transferred from the our website is not encrypted, allowing an attacker to read sensitive data in plaintext.	**High**	**6**
7	Malware/Trojan	A malicious software installed on any computer systems without any knowledge, and which can swipe up any sensitive data on the website and infect our system.	**High**	**7**
8	Insecure Design	Implementation defects of our website may lead to vulnerabilities that can be potentially be exploited .	**Medium**	**8**
9	Security Logging feature failures	Failure of logging features disable our website from detecting and responding to breaches and attacks in time.	**Medium**	**9**
10	Incompatible software and plugin update	Updating our current softwares/features/plugins/databases/firewall supporting our web application currently may conflict with the version of other software due to incompatibility.	**Medium**	**10**
11	Privilege Escalation	Adversary can gain elevated access to our website system and can obtain administration control over our web/database/firewall server.	**Low**	**11**

6. Threat Modelling

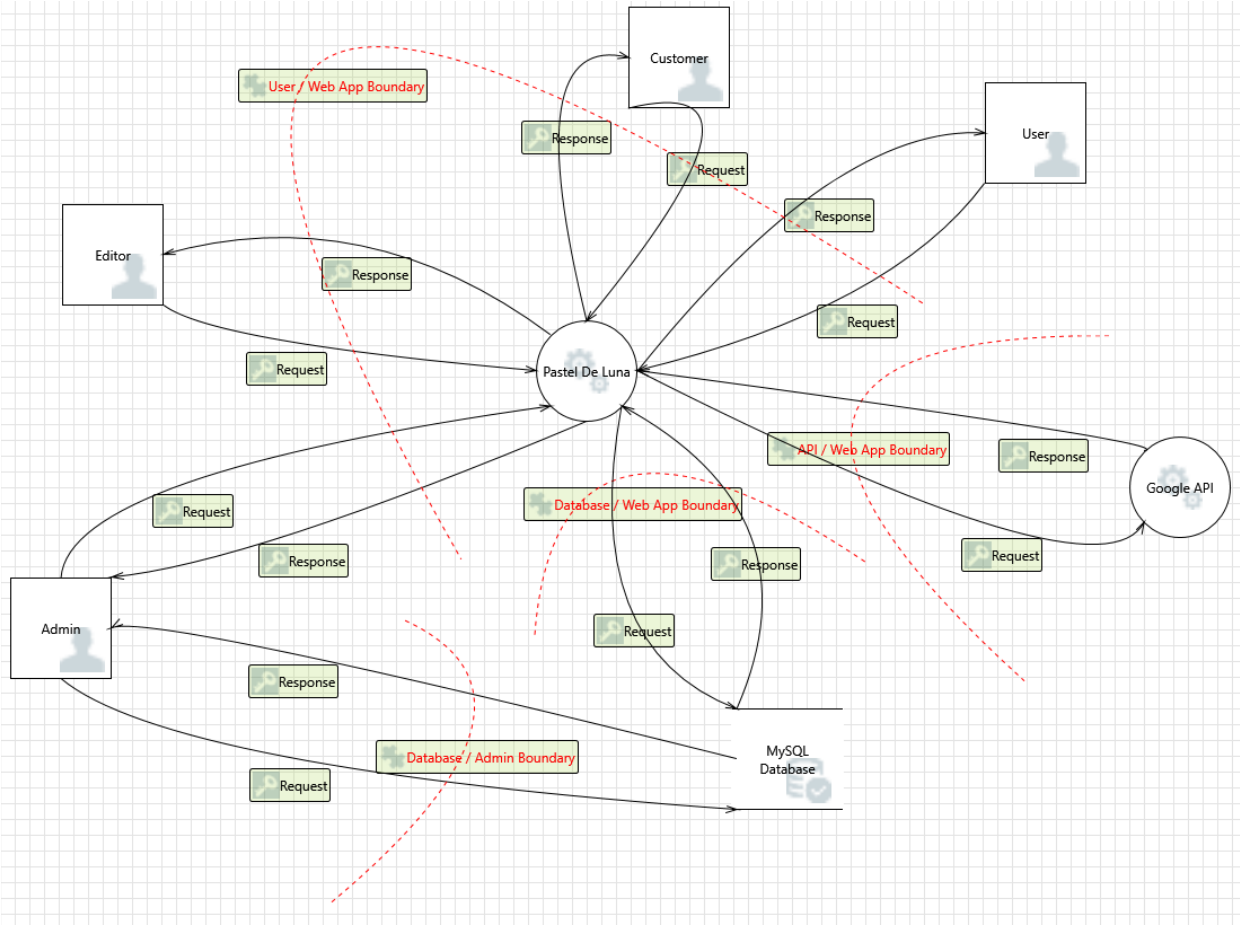

Figure 6: Threat Modeling Diagram

7. Attack Surface Analysis

Table 28: Attack Surface Analysis (Entry & Exit Points)

Entry Points			
S/N	**Affected Component**	**Potential Threats**	**Description**
Common pages			
1	Registration Page	SQL Injection	SQL injection can be used by an attacker to compromise the registration and the search bar where user inputs are required on our web application.
2	Nav Page(have Search Bar)	SQL Injection	
3	Login Page	SQL Injection	Attacker can perform SQL injection to login without actual credential.

Entry Points			
		Brute Force Crack Password	Attackers can employ trial and error and other applications to crack passwords, login credentials, and encryption keys through the username and password textbox.
		Session hijacking	Exploiting the website session control to gain access to a customer session token through unauthorised access to sensitive customer data.
4	Promotion Page	XSS Scripting	Adversaries can inject malicious code to the server and compromise the database, promotion page and home page.
5	Home Page		
6	Product detail Page	XSS Scripting	Adversaries can inject malicious code to the server and compromise the database and the product detail page.
		SQL Injection	SQL injection can be used by an attacker to compromise the product detail page input fields.
Customer			
7	Update Profile Page	XSS Scripting	Adversaries can inject malicious code to the server and compromise the database, update profile page and cart page.
8	Cart Page		
9	Payment Page	SQL Injection	Attacker can retrieve payment data belonging to other users and use it as his/hers through the input of the payment fields.
		Man In The Middle	Attacker can intercept the request and change the price of the products.
Editor			
10	Editor Dashboard	XSS Scripting	Adversaries can inject malicious code to the server and compromise the database, editor dashboard and product information page.
11	Product Information Page		
12	Edit Product Info Request Page	SQL Injection	SQL injection can be used by an attacker to compromise the edit product page field validation.
Administrator			
13	Administrator Dashboard	XSS Scripting	Adversaries can inject malicious code to the server and compromise the database and web system.

Exit Points			
S/N	**Affected Component**	**Potential Threats**	**Description**
Common pages			
1	Promotion Page	Cross-Frame Scripting	Data retrieved from the database to display on those pages. Adversary can inject malicious javascript with an payload iframe into the content data.
2	Home Page		
3	Product detail Page	XSS Scripting SQL Injection Cross-Frame Scripting	
Customer			
4	Profile Page	XSS Scripting	Data retrieved from the database to display on those pages. Malicious code able to inject into the content.
5	Cart Page		
6	Payment Page	Man-in-the middle attack	Data can intercept during the response and change the content of the data by adversary.
Editor			
7	Editor Dashboard	XSS Scripting	Data retrieved from the database to display on those pages.
8	Product information Page		
Administrator			
9	Administrator Dashboard	XSS Scripting	Data retrieved from the database to display on those pages.

8. Security Architecture

8.1 Physical System Architecture

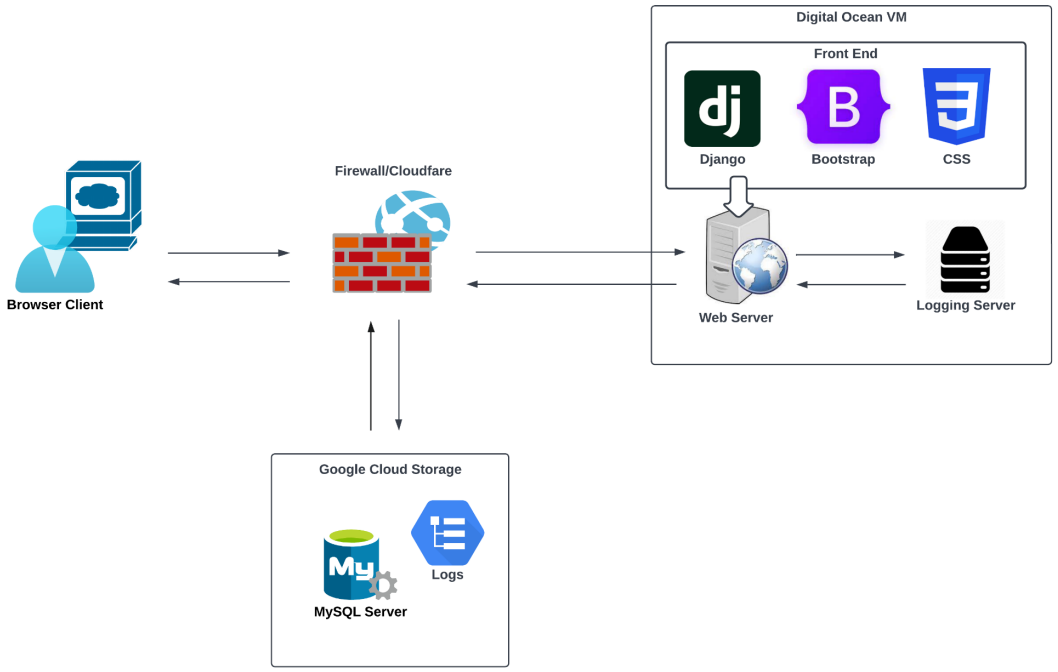

Figure 7: Physical System Architecture Diagram

8.2 Logical System Architecture

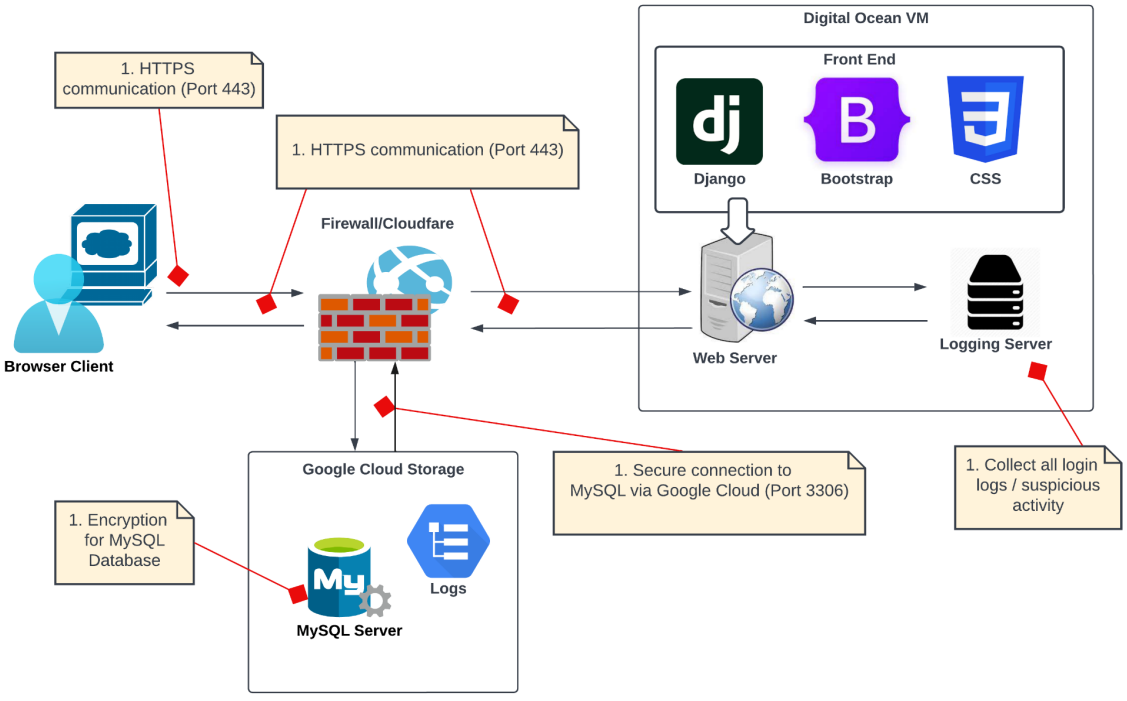

Figure 8: Logical System Architecture Diagram

9. Security Design

9.1 Threat identified and Solutions

Table 29: Threat identify in attack surface analysis

S/N	Threat Categories	Threat	Solution
1	Spoofing identity	Brute Force	Mixture of either solution will be sufficient to mitigate a bruteforce attack. The following are some implementations: 1. Setup a Password requirement to minimise successful brute force attempts: a. At least 8 characters b. A mixture of both uppercase and lowercase letters. c. A mixture of letters and numbers. d. Inclusion of at least one special character E.g ., ! @ # ?] 2. Setup an Account Timeout of 1 min after 5 wrong password tries within 15 seconds. This can be done by throttling the number of failed login requests for a given period. 3. Using Google ReCaptcha to minimise successful brute force attempts, as ReCaptcha will ensure a human is interacting with the web app. CAPTCHAs are particularly effective in stopping any kind of automated abuse, including brute-force attacks.
2	Spoofing Identity	Session-hijacking	If anyone that is logged in but idle for more than 5 min, there will be an auto log out.
3	Spoofing identity	Secure authentication failure	We have Set Up 2FA for Registration and Payment. This will help users to secure their account better.
4	Spoofing identity	Stolen Session ID	Under the profile Page, Users can view their Name,Address,Allergies and Payment details. For Payment details, the information we are going to show is only the last 4 digits. So the hacker couldn't steal the card details. During Payment, 2FA is required, so the user will be alerted if any purchase is made.
5	Tampering with data	Database got hacked and all the information got scrabble.	Backup database implemented will perform daily back-up of data from the main database to the backup database.
6	Potential Repudiation	Repudiation threats	Syslog/ log management is implemented to perform log management control for audit and incident response.

	of Data		
7	Information disclosure	Database Data leak	Information such as Password, will be encrypted with our encryption before storing into the database.
8	Information disclosure	Data sniffing	We will be using HTTPS only. HTTPS transfers data encrypted compared to HTTP that transfers data in plain text.
9	Information disclosure	SQL Injection	1. We will be using a Web Application Firewall (WAF). 2. We will have a logging system for the network flow. 3. Only password input fields can have special characters. 4. Proper Input validation and sanitization implemented for all input field
10	Denial of Service	Database Downtime	If the database is unable to connect for more than 10 seconds, the web application will auto link up to the backup database instead.
11	Denial of Service	Database got deleted	If the database gets deleted, the web application will auto link up to the backup database instead. The backup will be duplicated to ensure redundancy and increase service reliability.
12	Denial of Service	DDOS	Our web server implemented WAF and access control list to filter which traffic can reach our server. Reducing the attack surface area was also considered in our application.
13	Elevation of Privileges	Privilege Escalation	Our web application is set to 4 privilege, 1. Normal User a. Browse Website Only 2. User with a registered account a. Browse Website b. Add Item to cart c. Make Payment 3. Editor(Seller) a. Browse Editor Dashboard b. Send a request to edit product information that they are selling 4. Administrator a. View request sent by Editor on the Admin Dashboard b. Change the status of the Editor Request. (Pending/Approved/Denied) So if the attacker is able to gain administrator access, there is nothing much they can do other than view the changes requested by the editor.

9.2 Database schema

The team is using a relational database (MySQL) to store the e-commerce information. As such, there is a need to ensure relationships integrity (referential integrity) between tables that are defined in our database. To achieve that, the team associated each table with a primary key as seen in Figure 9. This allows each table to uniquely identify each row in the table and to be easily joined across other tables through the use of foreign keys.

However, the threat considered when using primary and foreign keys is that when not properly defined, the deletion of the primary key row can have an effect on the foreign key row (i.e. children become orphans when parents are deleted). For example, when a Product_Detail is deleted the associated Promotion to the Product_Detail should not exist. To achieve that, the team can use the ON DELETE CASCADE option to specify the child to be deleted when the parent row does not exist or it must be indicated as null. Hence, by ensuring referential integrity improve data traceability and security by minimising data corruption.

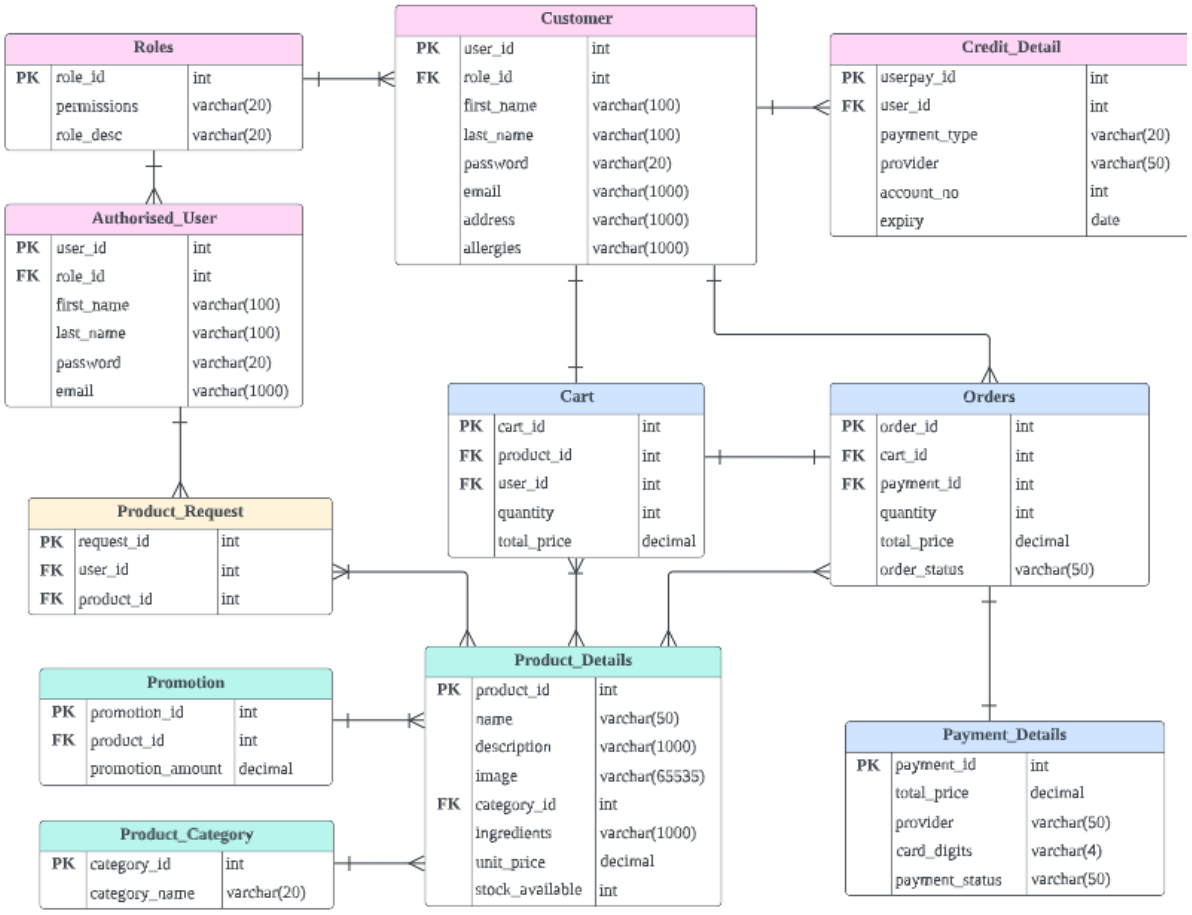

Figure 9: ERD Diagram

APPENDICES

APPENDIX A: USE CASE DESCRIPTIONS

Table 30: User use case - Browse the Website

Use Case ID - Name	UC1 - Browse the Website
Actor	User
Description	The use case allows the user to browse the website and look through what Pastel De Luna can offer.

Table 31: User use case - Search for mooncake

Use Case ID - Name	UC2 - Search for mooncake
Actor	User
Description	The use case allows the user to search for the mooncake using the search bar so that they are able to find the product more efficiently.

Table 32: User use case - Register for account

Use Case ID - Name	UC3 - Register for account
Actor	User
Description	The use case allows users to register for a customer account so that they are able to login to continue their orders.

Table 33: User use case - Login/ Logout

Use Case ID - Name	UC4 - Login/Logout
Actor	User
Description	The use case allows the user to perform login and logout from the website.

Table 34: Customer use case - Manage Profile

Use Case ID - Name	UC5 - Manage Profile Page
Actor	Customer
Description	The use case will allow customers to manage their profile page (create, retrieve, update) so that their personal details are always updated.

Table 35: Customer use case - Manage mooncakes from the Cart

Use Case ID - Name	UC6- Manage mooncakes from the Cart
Actor	Customer
Description	The use case will allow customers to manage their cart items (create, retrieve, update, delete) so that they can customize their online order.

Table 36: Customer use case - Make payment

Use Case ID - Name	UC7 - Make payment
Actor	Customer
Description	The use case will allow customers to make a purchasement of the cart order so that they can pay online for their orders.

Table 37: Customer use case - Receive Confirmation Email

Use Case ID - Name	UC8 - Receive Confirmation Email
Actor	Customer
Description	The use case will allow customers to receive a confirmation email once they have successfully paid for their orders.

Table 38: Customer use case - View Payment History

Use Case ID - Name	UC9 - View Payment History
Actor	Customer
Description	The use case will allow customers to view their payment history so that they can track and review their payment.

Table 39: Customer use case - View Order Status

Use Case ID - Name	UC10 - View Order Status
Actor	Customer
Description	The use case will allow customers to view their order status so that they understand how their order has been processed throughout.

Table 40: Administrator use case - Create Editor Account

Use Case ID - Name	UC11 - Create Editor Account
Actor	Administrator
Description	The use case allows the administrator to create an editor account through

	the database.

Table 41: Administrator use case - Delete Editor Account

Use Case ID - Name	UC12 - Delete Editor Account
Actor	Administrator
Description	The use case allows the administrator to delete the editor account through the database.

Table 42: Administrator use case - Manage Product Request

Use Case ID - Name	UC13 - Manage Product Request
Actor	Administrator
Description	The use case allows the administrator to receive a product change request given by the editor so that they can make the necessary changes on the database.

Table 43: Editor use case - Send Request to manage the Product

Use Case ID - Name	UC14 - Send Request to manage the Product
Actor	Editor
Description	The use case allows the editor to send a product change request to the administrator so that the product catalogue content can change accordingly.

Table 44: Customer use case - Browse the Website

Use Case ID - Name	UC15 - Browse the Website
Actor	Customer
Description	The use case allows the user to browse the website.

Table 45: Customer use case - Search for mooncake

Use Case ID - Name	UC16 - Search for mooncake
Actor	Customer
Description	The use case allows the user to search for the mooncake using the search bar so that they are able to find the product more efficiently.

Table 46: Customer use case - Login/Logout

Use Case ID - Name	UC17 - Login/Logout
Actor	Customer
Description	The use case allows the user to perform login and logout from the website.

Table 47: Administrator use case - Login/Logout

Use Case ID - Name	UC18 - Login/Logout
Actor	Administrator
Description	The use case allows the user to perform login and logout from the website.

APPENDIX B: USER ACCESS MATRIX

Table 48: Access matrix for user, customer, editor, administrator

Web Application				
	Actors			
Types of Data	**User**	**Customer**	**Editor**	**Administrator**
User Profile	-	**CRUD**	-	-
Product details	**R**	**R**	**CRUD**	-
Editor Request	-	-	**CRUD**	**R**

Database				
	Actors			
Types of Data	**User**	**Customer**	**Editor**	**Administrator**
User information (encrypted)	-	**CRUD**	-	-
Product Details	-	-	-	**CRUD**
Editor information	-	-	-	**CRUD**

APPENDIX C: PROJECT TASK BOARD

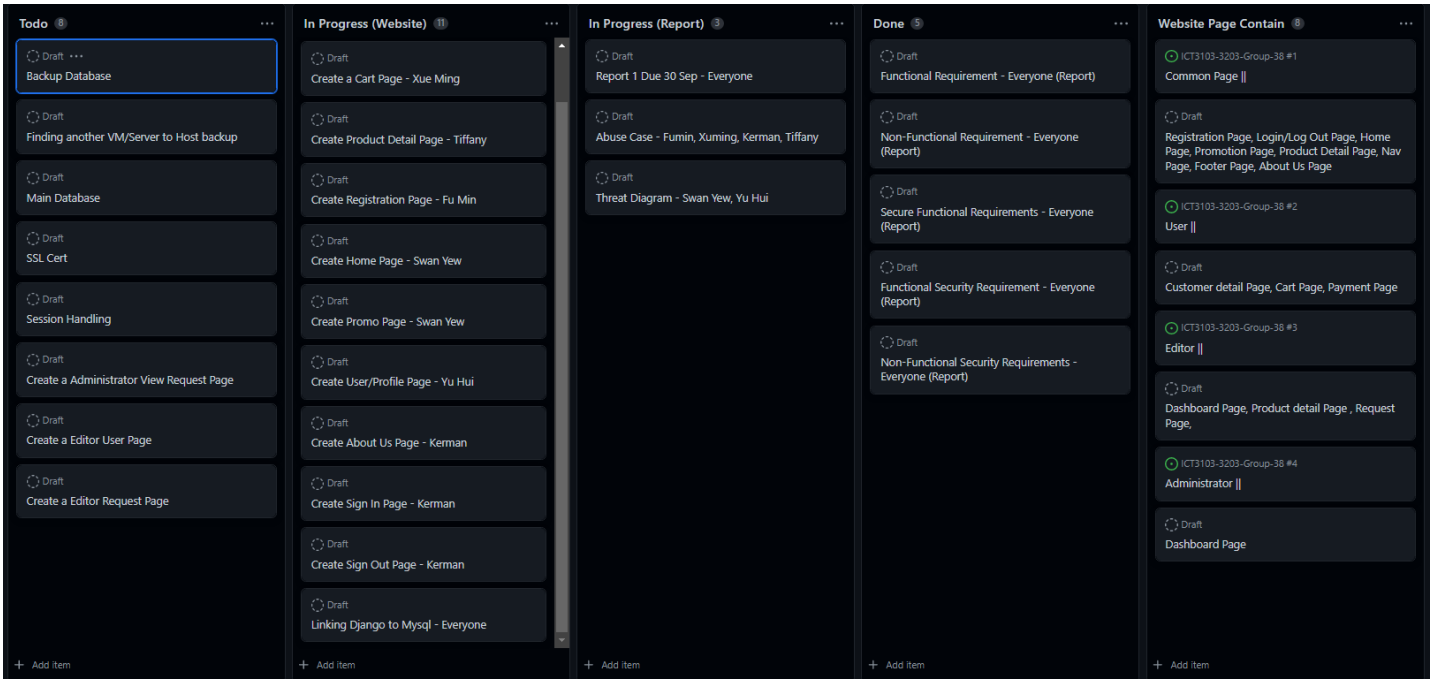

Figure 10: Task Board