

**CCS6344 T2510 Assignment 1 Submission**

**Group Name: Group 12**

**Project** **Title : Bakery Ordering System**

**Lecturer Name: Dr. Navaneethan A/L C.Arjuman**

**Prepared By:**

|  |  |
| --- | --- |
| **Student ID** | **Name** |
| 1211112284 | Nur Dania Iman Binti Desman Desa |
| 1211101800 | Tan Jia Jin |
| 1211101775 | Lam Yuet Xin |

YouTube link:

GitHub link:

Contents

[Task 1 - Preparation of the proposal 3](#_Toc198371881)

[1.1 Objective of the project. 3](#_Toc198371882)

[1.2 Design and Implementation of the Application 4](#_Toc198371883)

[1.3 Hardware and software to develop the application 4](#_Toc198371884)

[1.4 System design and database design for this application 5](#_Toc198371885)

[1.5 Secured database using the SQL database system 5](#_Toc198371886)

[Task 2 - Implementation of the application using SQL Database 6](#_Toc198371887)

[2.1 System Design 6](#_Toc198371888)

[2.2 Whole Workflow Application Screenshot 6](#_Toc198371889)

[2.3 Security Measures Implementation 6](#_Toc198371890)

[2.4 Input Validation 6](#_Toc198371891)

[Task 3 Threat Modelling 6](#_Toc198371892)

[Task 4 PDPA 2010 7](#_Toc198371893)

[Task 5 Security Measures Implementation 7](#_Toc198371894)

# Task 1 - Preparation of the proposal

## 1.1 Objective of the project.

The aim of this project is to design, develop, and secure a web-based application using the conventional SQL database system in line with comprehensive security best practices and processes for regulatory compliance with PDPA 2010 regulations. An application is created to effectively demonstrate the implementation of database security measures properly to make it secure against common vulnerabilities and threats.

Threat analysis will be done by applying STRIDE and DREAD.

Validation of protection will be tested through penetration testing. Correct data handling procedures under PDPA 2010 will be implemented. The project will present project documentation, including a technical report and presentation.

* Build a web application with proper authentication mechanism
* Imposing security controls on general threats like SQL injection attacks
* An analysis of threats using the STRIDE and DREAD framework
* Working on data handling processes conforming to the PDPA 2010 regulations
* Validation of security through penetration tests
* Preparation of all professional documentation and presentation

## 1.2 Design and Implementation of the Application

The Bakery Ordering System is a web-based application designed to allow customers to conveniently place food orders online. The system emphasizes both security and user experience, ensuring that interactions on both the customer and administrator sides are safe, smooth, and user-friendly.

The key features of the application include:

* User Management:

Customers can create accounts and log in securely. Input validation is implemented to prevent security vulnerabilities and protect against automated login attempts, ensuring secure and reliable access.

* Menu and Products:

Customers can browse a clearly displayed menu that lists all available food items along with their prices before placing an order.

* Staff Management:

Staff members can also create accounts and log in securely. Similar input validation measures are in place to prevent security vulnerabilities and unauthorized access.

* Order Management System:

Users can place and manage their orders efficiently through the system.

* Secure Payment System:

Payments are handled securely using encryption to protect payment information. Input validation is also implemented to prevent fraudulent or incorrect data entries, maintaining the integrity and security of every transaction.

## 1.3 Hardware and software to develop the application

1. Programming Language: PHP for backend, HTML/CSS/JavaScript for frontend.
2. Database: MSSQL for data storage. SSMS for database management.
3. Operating System: Windows 11.
4. Server: XAMPP Local Server

## 1.4 System design and database design for this application

System Design

The system is designed using a multi-layered approach:

1. **Application Layer:**

This is the backend of the system, responsible for handling user requests, executing business logic, and managing the communication between the user interface (presentation layer) and the database (data layer).

1. **Presentation Layer:**

This is the part of the system that users interact with directly. It includes web pages, forms, and other elements of the user interface that allow users to browse, input, and view information.

1. **Data Layer:**

This layer uses an MSSQL database to store all necessary data for the application. It includes user accounts, product details, staff records, order histories, and payment information.

Database Design

The database includes the following tables:

1. Customers: To store the customer’s personal details.
2. Products: To store product details.
3. Staff: To store the staff’s personal details.
4. Orders: To store order receipt.
5. Order Items: To store order details placed by customers.
6. Payment: To manage payment details.

## 1.5 Secured database using the SQL database system

To secure the database, the following measures are implemented:

a. **Authentication**:  
Before accessing the database, users must prove their identity by entering a valid username and password. These credentials are checked against those stored in the system to confirm the user is legitimate.

b. **Authorization**:  
Once authenticated, users are only allowed to access the data they are permitted to see or change. This is based on their specific roles and access levels within the system.

c. **Data Encryption**:  
Sensitive data stored in the database is encrypted, meaning it’s converted into a secure format that can’t be easily read by unauthorized users.

d. **Password Hashing**:  
Passwords are converted into a secure hashed format before being stored in the database. This ensures that even if the database is compromised, the original passwords remain protected.

e. **Data Masking**:  
Sensitive details, such as credit card numbers, are partially hidden or masked in the database so that the full information is not visible to unauthorized users.

f. **Input Validation**:  
All data entered by users is checked to make sure it’s in the correct format and doesn’t contain harmful code or malicious input, helping to prevent security threats like injection attacks.

# Task 2 - Implementation of the application using SQL Database

## 2.1 System Design

The Bakery Ordering System is an online platform that allows users to conveniently place food orders. The system is designed to support three main roles: **Admin**, **Staff**, **User**,and **Guest**.

* **User Role**:  
  Users can log in to the website and view the menu, which displays available food items and their prices. After selecting their desired items, users proceed to the order form to place their order. Once the order is submitted, payment is made manually at the counter, where staff will update the system with the user's credit card number for processing. When all steps are completed successfully, the user's order is confirmed.
* **User Role (Guest)**:

Guest can access in to the website and view the menu which displays only available food items without pricing. If guests want to make an order or discover pricing, they must create an account by directing to sign up page from “Join Us!” button (guest order page) or go to sign up.

* **Admin Role**:  
  Admins can log in to the system to access and manage various aspects of the platform. They can view all orders, monitor payment history, manage products, and access information about both customers and staff members.
* **Staff Role**:  
  Staff members have access similar to the admin, with the ability to view orders, manage products, and see customer information. However, they **do not** have access to view or manage staff details.

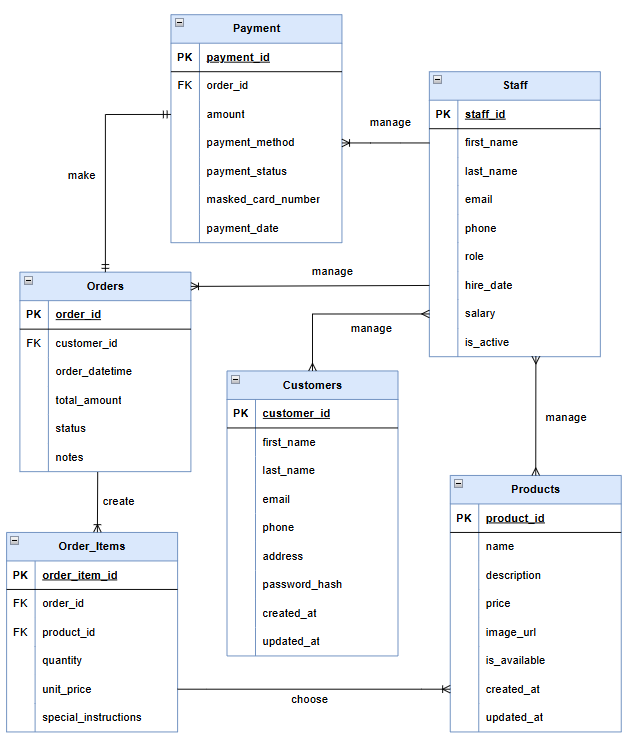


Figure 2.1.0 Bakery Order System Entity Relational Diagram.

## 2.2 Whole Workflow Application Screenshot

1. Guest

* Log in page

A screenshot of a login form

AI-generated content may be incorrect.

Figure 2.2.0 Users can continue as guest.

* Menu

A close up of food

AI-generated content may be incorrect.Figure 2.2.0 Users can Home page.

* Menu

A collage of food on a screen

AI-generated content may be incorrect.

Figure 2.2.0 Users can view menu without price.

* Order

A screenshot of a computer screen

AI-generated content may be incorrect.

Figure 2.2.0 Users can view menu without price.

* Sign up

A sign up form with white text

AI-generated content may be incorrect.

Figure 2.2.0 Users is directed to sign up page.

1. Customer

* Register / Sign Up

A screen shot of a sign up form

AI-generated content may be incorrect.

Figure 2.2.0 Users can register their account by inserting their details including password.

* Login

A screenshot of a login form

AI-generated content may be incorrect.

Figure 2.2.1 Users can log in their account by inserting their email and password.

* Home

A close up of food

AI-generated content may be incorrect. Figure 2.2.2 User access to Crumby Café home page.

* Menu

A screenshot of a food photo

AI-generated content may be incorrect.

Figure 2.2.3 Users can view menu.

* Order

A computer screen shot of a computer screen

AI-generated content may be incorrect.

Figure 2.2.4 Users can create an order.

* Make Payment (Staff update manually in database)

A screenshot of a computer

AI-generated content may be incorrect.

Figure 2.2.5 Users need to make payment at the counter.

* Order history

A screenshot of a computer

AI-generated content may be incorrect.

Figure 2.2.5 Users can view order history.

1. Admin

* Admin dashboard
  + Customer Information A screenshot of a computer

    AI-generated content may be incorrect.

Figure 2.2.4 Admin can see customer details.

* Recent Orders

A screenshot of a computer

AI-generated content may be incorrect.Figure 2.2.5 Admin can see past ordered created by customer.

* Payment HistoryA screenshot of a computer

  AI-generated content may be incorrect.

Figure 2.2.6 Admin can see history payment made by customer.

* Order ItemA screenshot of a computer

  AI-generated content may be incorrect.

Figure 2.2.7 Order Items or foods by customer.

* Staff Information

A screenshot of a computer

AI-generated content may be incorrect.

Figure 2.2.8 Admin can see staff details.

* Product Management

A screenshot of a computer

AI-generated content may be incorrect.

Figure 2.2.9 Admin can manage the stock of the product.

1. Staff

* Dashboard
  + Customer Information A screenshot of a computer

    AI-generated content may be incorrect.

Figure 2.2.10 Staff can see customer details.

* Recent Orders

A screenshot of a computer

AI-generated content may be incorrect.Figure 2.2.11 Staff can see past ordered created by customer.

* Payment HistoryA screenshot of a computer

  AI-generated content may be incorrect.

Figure 2.2.12 Staff can see history payment made by customer.

* Order ItemA screenshot of a computer

  AI-generated content may be incorrect.

Figure 2.2.13 Order Items or foods by customer.

* Product

A screenshot of a computer

AI-generated content may be incorrect.

Figure 2.2.14 Staff can manage the stock of the product.

## 2.3 Security Measures Implementation

**Authentication:**

**Authorization :**

|  |  |  |
| --- | --- | --- |
|  | Staff | Admin |
| Dashboard | Checkmark with solid fill | Checkmark with solid fill |
| Staff Information | Close with solid fill | Checkmark with solid fill |
| Product Management | Checkmark with solid fill | Checkmark with solid fill |

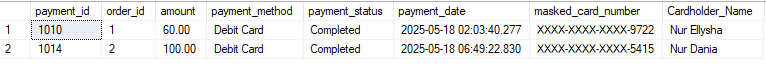
**Hashing :** Hashing is used to secure data stored in a database, such as user passwords. When a new user creates an account, their password is converted into a hashed format before being stored. This helps maintain data integrity because the original password cannot be easily retrieved or read by unauthorized users. Since hashing generates a unique value, even a small change in the original data will result in a completely different hash. This makes it easier to detect if the database has been tampered with.

A screenshot of a computer

AI-generated content may be incorrect.

Figure 2.3.4 Masking Card Number.

**Masking :** The web application employs masking as a security technique to protect sensitive information, as shown in Figure 2.3.4. When displaying credit card numbers, only the final four digits (9722) remain visible, while the first twelve digits are replaced with 'X' characters (XXXX XXXX XXXX 9722). This partial concealment significantly reduces the risk of unauthorized access and potential misuse of card information, thereby preserving the confidentiality of this critical security element.

Figure 2.3.4 Masking Card Number.

**Encryption :** Figure 2.3.5 demonstrates encryption implementation for protecting sensitive data, specifically the Cardholder\_Name field. The system converts the plaintext name into ciphertext using the aes-256-cbc cryptographic algorithm with a specific encryption key. This transformation ensures that the information remains secure and can only be accessed by authorized parties who possess the correct decryption key. This encryption strategy is essential for safeguarding personal and financial information from unauthorized access and security breaches, particularly in banking contexts. By implementing this encryption method, our website maintains compliance with data protection requirements while minimizing the threat of data theft, as any intercepted information would remain unusable without the proper decryption tools.

Figure 2.3.5 Encryption of Cardholder Name.

**Decryption -**  Only authorized parties who possess the correct decryption key can accessed the original name.

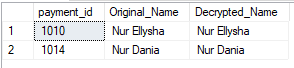


Figure 2.3.6 Decryption of Cardholder Name.

**Input Validation:**

A form input validation message is triggered if the First Name field is left empty.

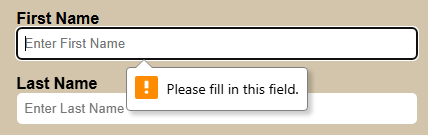


Figure 2.3.7 First Name Validation.

email input validation error prompting the user to include an '@' symbol, if is not a valid email format.

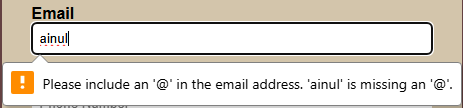


Figure 2.3.8 Email Validation.

password input validation error indicating that the entered password must be at least 8 characters long.

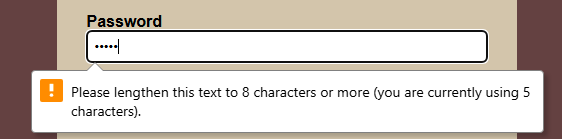


Figure 2.3.9 Password Validation.

# Task 3 Threat Modelling (STRIDE AND DREAD)

|  |  |
| --- | --- |
| Spoofing Identity | To prevent spoofing:   * Strong user authentication with secure login. * Input validation to block fake credentials. |
| Tampering with data | To prevent tampering with data:   * Role-based access control (RBAC) ensures only authorized users can modify specific data. |
| Repudiation | To prevent repudiation:   * Restrict access to sensitive data using role-based permissions and database-level access controls. * Encrypt sensitive data to protect it from unauthorized access or leaks, even if the system is breached. |
| Information Disclosure | To prevent information disclosure:   * Data encryption for sensitive data (e.g., credit card numbers). * Data masking for confidential information. |
| Denial of Service | To prevent denial of service:   * Input validation prevents system overload. |
| Elevation of Privileges | To prevent elevation of privileges:   * RBAC implementation to restrict access. * Regular review and updates to permissions and roles. * No direct user access to admin roles. |

|  |  |  |
| --- | --- | --- |
| Character | Acronym for | Rating |
| D | Damage potential | 1 = minimal damage, 10 = catastrophic vulnerability |
| R | Reproducibility | 1 = hard to reproduce, 10 = easily repeatable, e.g., by bots |
| E | Exploitability | 1 = hard to exploit, 10 = easily exploited vulnerability |
| A | Affected Users | |  | | --- | | Divide % affected users by 10 and round (e.g., 80% → 8) | |
| D | Discoverability | 1 = obscure vulnerability, 10 = easy to find (e.g., SQL injection) |

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Stride Category | Threat Detection | D | R | E | A | D | Threat  Rating |
| S | Customer or attacker logs in as staff/admin | 4 | 3 | 6 | 5 | 6 | 4.8 |
| T | Order manipulation (price/quantity modification) | 3 | 4 | 5 | 4 | 5 | 4.2 |
| R | Customer denies payment or staff denies action | 2 | 2 | 3 | 3 | 4 | 2.8 |
| I | Card info or order history leak | 5 | 3 | 5 | 6 | 6 | 5.0 |
| D | Too many requests prevent new orders | 7 | 6 | 6 | 7 | 6 | 6.4 |
| E | Customer gains admin/staff access | 6 | 4 | 6 | 5 | 5 | 5.2 |

# Task 4 PDPA 2010

**Categorize of Personnel According to PDPA 2010:**

**1. Data Users (Data Controllers)**

**Definition**: Organizations or individuals who collect, store, and process personal data, and have control over or authorize the processing of any personal data.

**Responsibilities**:

* Ensure compliance with data protection principles
* Implement appropriate security measures
* Inform data subjects about data processing activities
* Respond to data subjects' requests regarding their personal data

**2. Data Subjects**

**Definition**: Individuals whose personal data is being collected, stored, or processed.

**Rights**:

* Right to be informed about the processing of their data
* Right to access their personal data
* Right to correct inaccurate data
* Right to withdraw consent for data processing
* Right to prevent processing likely to cause damage or distress

**3. Data Processors**

**Definition**: Entities or individuals that process data on behalf of the data user.

**Responsibilities**:

* Process data only on the instructions of the data user
* Implement appropriate technical and organizational measures to protect personal data
* Assist the data user in fulfilling data subject requests

**4. Data Protection Officer (DPO)**

**Definition**: A designated individual responsible for overseeing data protection strategy and compliance.

**Responsibilities**:

* Ensure the organization's compliance with the PDPA
* Conduct regular data protection audits
* Serve as a point of contact for authorities and data subjects
* Provide training on data protection

**5. Internal Audit and Compliance Team**

**Definition**: Team responsible for ensuring internal policies and procedures comply with the PDPA.

**Responsibilities**:

* Conduct regular audits to ensure data protection practices are followed
* Review and update data protection policies
* Provide recommendations for improving data protection measures

By categorizing personnel in this way, organizations can ensure that all roles and responsibilities related to data protection are clearly defined and managed according to the requirements of the PDPA 2010.

**Mapping Data Lifecycle to PDPA 2010 Requirements:**

**1. Collection Stage**

**PDPA 2010 Requirements**:

* **Notice and Choice Principle:** Data subjects must be informed about the processing of their personal data through written notice.
* **General Principle:** Personal data must be processed lawfully and shall not be processed without the consent of the data subject.
* **Purpose Limitation**: Collect data only for specified, legitimate purposes.
* **Data Minimization**: Collect only the data necessary for the specified purposes.

**2. Storage Stage**

**PDPA 2010 Requirements**:

* **Security Principle:** Take practical steps to protect personal data from loss, misuse, modification, unauthorized access, or disclosure.
* **Retention Principle:** Personal data shall not be kept longer than necessary.
* **Data Integrity**: Ensure that the data stored is accurate, complete, and kept up to date.

**3. Usage Stage**

**PDPA 2010 Requirements**:

* **Disclosure Principle:** No disclosure without consent
* **Purpose Principle:** Data should only be processed for the specified purposes
* **Access Control**: Restrict access to personal data to authorized personnel only
* **Data Quality**: Ensure the data remains accurate, relevant, and up to date for its intended use

**4. Sharing and Disclosure Stage**

**PDPA 2010 Requirements**:

* **Disclosure Principle:** Personal data shall not be disclosed without consent
* **Third Party Agreements**: Ensure third parties processing personal data have adequate protection measures
* **Data Transfer Conditions**: Follow specific conditions for transferring data to third parties
* **Consent for Sharing**: Obtain consent before sharing data with third parties

**5. Retention and Disposal Stage**

**PDPA 2010 Requirements**:

* **Retention Principle:** Data shall not be kept longer than necessary
* **Data Integrity Principle:**  Take reasonable steps to ensure that personal data is accurate, complete, not misleading, and kept up to date
* **Secure Disposal**: Ensure secure disposal of data when no longer needed

**6. Access and Correction Stage**

**PDPA 2010 Requirements**:

* **Access Principle (Section 12)**: Data subjects have the right to access their personal data
* **Correction Principle (Section 13)**: Data subjects have the right to correct their personal data
* **Response to Requests**: Respond to data access and correction requests within a reasonable time frame

**7. Audit and Review Stage**

**PDPA 2010 Requirements**:

* **Compliance Audits**: Regularly conduct audits to ensure compliance with PDPA requirements
* **Policy Review**: Periodically review and update data protection policies
* **Incident Management**: Have procedures for managing and reporting data breaches

# Task 5 Security Measures Implementation

**Role and SQL User**

1. Bakery\_customer

* Can view products and place orders.

1. Bakery\_staff

* Can manage orders, update product inventory, and process payments.
* Read/write access to operational data, but no access to HR or admin-level settings.

1. Bakery\_admin

* Has full control over bakery operations
* Can create or modify bakery\_staff and customer accounts.
* Cannot access or modify system-level or database owner settings.

1. Hr

* Can create, update, or delete staff records.
* No access to customer data or system/database configuration.

1. Dbowner

* Has full control over the database schema and objects (tables, views, procedures).
* Can grant/revoke permissions, backup/restore database, and manage structure.

1. Sysadmin

* Highest level access – can manage server settings, all databases, users, and roles.
* Can perform any task across the SQL Server instance, including role assignments.

1. Guest

* Has limited access – can view certain public data only.
* Cannot make any changes or access restricted data.