**[1. Introduction project 2](#bookmark1)**

**[2. Project purpose 2](#bookmark2)**

**[3. Functional design 3](#bookmark3)**

[3.1 Administrator 5](#bookmark4)

[3.2 Customers 13](#bookmark5)

**[4. Development approach 17](#bookmark6)**

**[5. Flowchart 20](#bookmark8)**

**[6. Contribution 20](#bookmark9)**

**1. Introduction Project**

QuickMart Online Shopping System is a small, convenient and personalized online mall

with two pages, management personnel and customers, which is convenient for

different audiences. The project was inspired by providing an online marketplace for

smaller stores, which not only increased customer loyalty, but also allowed employees to adjust prices and storage by looking at product trends through data visualization. In this document, the background and objectives, technical selection, functions and

flowchart of the project will be described in detail.

**2. Project purpose**

The project mainly serves small shopping malls, so the project objectives include

1. Increase in sales, number of orders, sales rate

2. Increase in the number of loyal users, visits and registrations, and active users

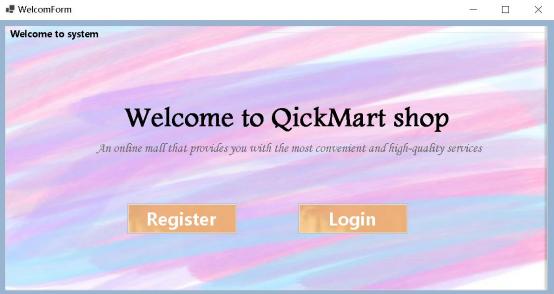
3. Efficient inventory management, reduce inventory, waste, backlog and outstock

4. The proper discount can not only ensure that the mall achieves high profitability, but also attracts new users to buy goods

**3. Functional design**

**Welcome page**

Description: Users open the first page displayed by the program and can jump to the registration and login pages.



**Registration page**

Description: Users (regardless of role) can register a new account, need to fill in the name phone number role birthday address password field, the name cannot be the same as the name in the existing database, the phone number needs to follow the 10-digit number zero four, the birthday must be before the current date, the

password needs to be the same twice, if the rules cannot be followed, an error window will pop up. The correct new account information will be stored in the database.



**Login page**

Description: The user needs to fill in the name and password that match the

database storage information, and select the correct role, if the selection is wrong, there will be an error message. The page also contains buttons to return to the

welcome page and buttons to exit the system.



The system matches the window of the corresponding role based on the role selected by the user.

**3.1 Administrator**

**Admin Home Page**

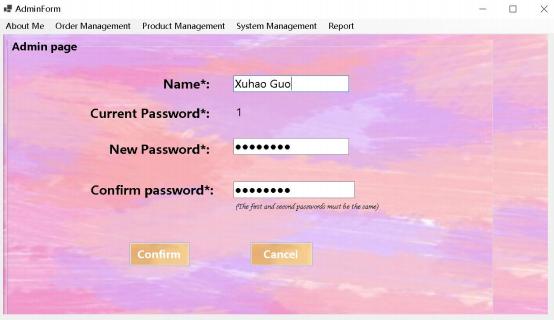


**Modify Password page**

Description: Enter this window will see the name and password of the current user,

the user can modify the name and confirm the new password, if the name is the same as the name of the current database, the two passwords are entered differently, there will be an error prompt, press the confirm button to update the database and press

the cancel button to return to the main page.



**Admin Information page**

Description: Displays the pre-stored administrator information in the database,

including ID, name, role, phone number, password, birthday, address, photo, and users can update the current photo by themselves.



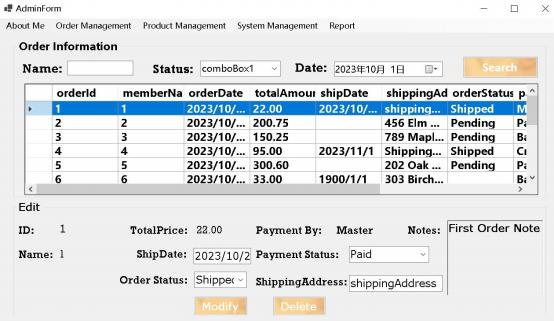
**Order management page**

Description: The page will display all the order details in the system, click a piece of

data to modify the delivery time, payment method, order status and delivery address, and search for the corresponding order according to the name, order status and

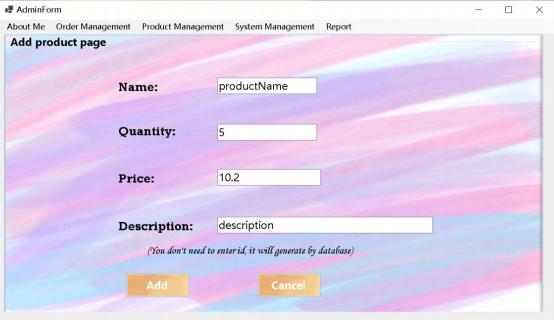
delivery

date



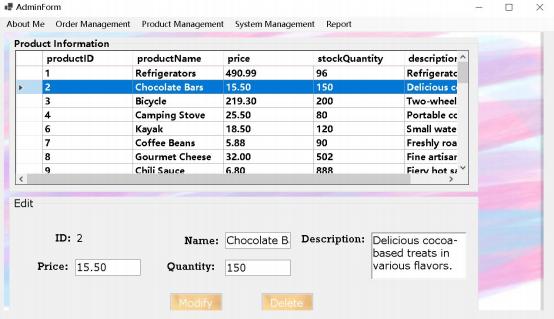
**Product addition page**

Description: Administrators can add a new product to fill in the name, quantity, price, and description of the product.



**Product management page**

Description: All product details in the system will be displayed, including product ID, name, price, description, inventory, and the administrator can modify all information except the product ID, and can also delete the product



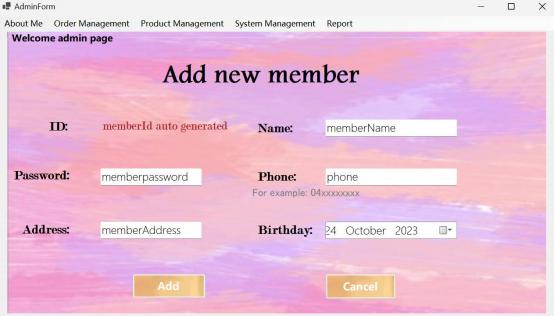
**Add a new member page**

Description: Administrator can add a new member member ID will automatically

generate the need weather name, password, phone number address, and birthday

field, birthday must be earlier than the current time, the phone number also needs to match the 10 digits starting with zero four, otherwise there will be an error prompt

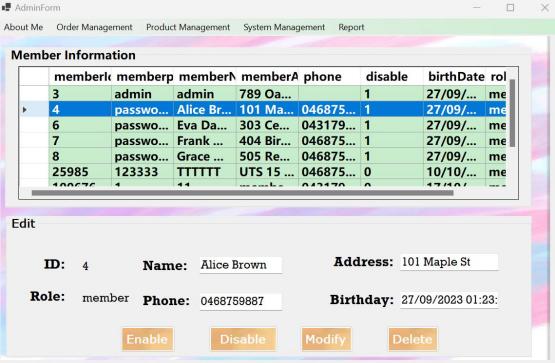
Press the Add button, data will be added to the database



**Modify membership page**

Description: The page will display all the member information in the database clicks, each piece of data will be modified in the following modification box, the

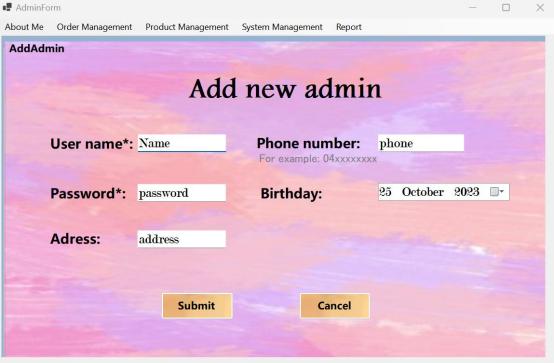
administrator can also delete the member information or disable the member to use.



**Add an admin page**

Description: Administrators can add new administrators need to fill in user name,

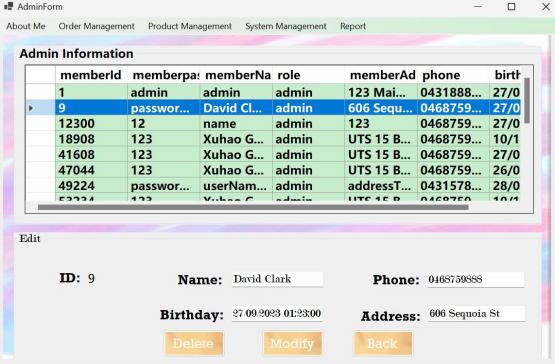
phone number, birthday password and address Press the submit button, the new data will be added to the database.



**Modify the Administrator page**

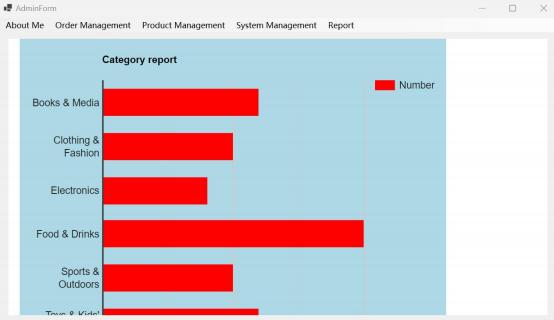
Description: All the administrator information stored in the database will be displayed, and the new data will be transferred to the database after clicking a certain detail to

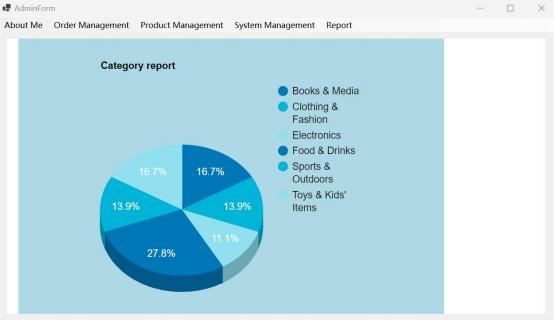
modify and delete the data and pressing the modify button



**Data analysis page**

Description: The number of each product category will be visualized in a bar chart, and the proportion of each quantity in the database can be viewed through a pie chart.

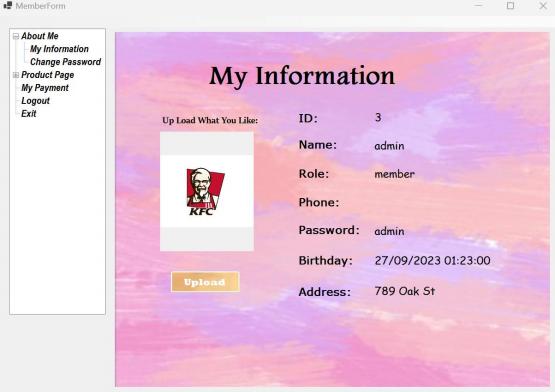




**3.2 Customers**

**Customer information page**

Description: Displays the information of the current customer in the database, and the customer can modify the picture

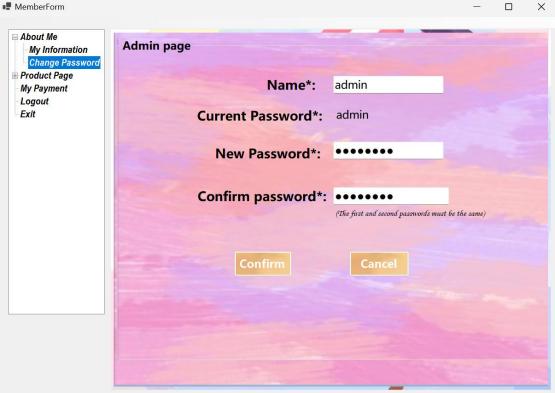


**Modify Password page**

Description: Enter this window will see the name and password of the current user,

the user can modify the name and confirm the new password, if the name is the same as the name of the current database, the two passwords are entered differently, there will be an error prompt, press the confirm button to update the database and press

the cancel button to return to the main page.



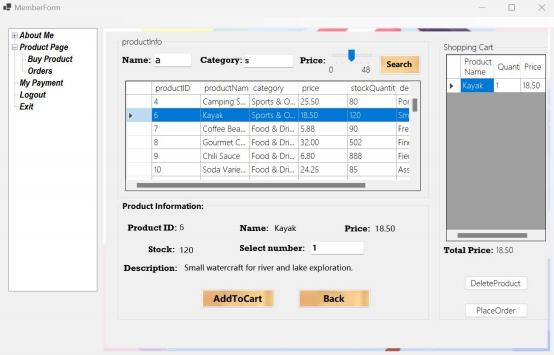
**Purchase product page**

Description: Users can view all products in the current store, and view their category price inventory and description, users can choose the quantity purchased, users can

search for the corresponding products by searching the name search category and

price range, users can add goods to the shopping cart, and view the current total price in the shopping cart, users can delete goods in the shopping cart Users can click Send Order, when the user clicks, the inventory quantity on the Send Order page will be

reduced.

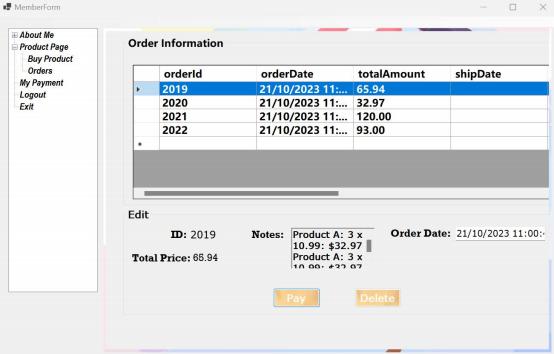


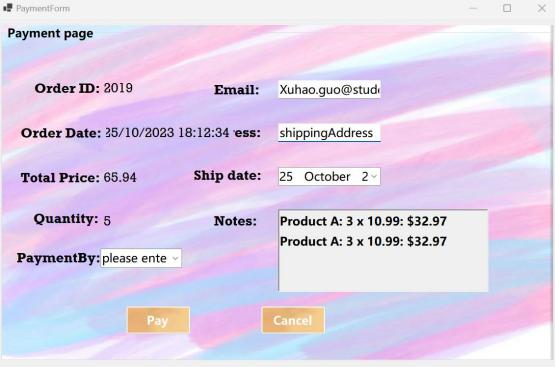
**Orders page**

Description: Users can view all orders they have sent, including the date the order was

sent, the quantity in the details of the order, and date of delivery, if the user clicks the payment button, it will be taken to the payment page, and if the user clicks the delete button, the order data will be deleted in the database. After paid, it will send an

confirmation email!









**4. Development approach**

Due to the possibility of continuous evolution and iteration of the commodity purchase and management system in response to business needs, we have opted for Agile approach, as it allows for rapid iteration and adaptability. We have divided the development process into multiple iterations, each delivering a potentially shippable product version. Each iteration introduces new features or enhances existing ones. In each iteration version, we implement automated unit testing and integration testing

using NUnit, including parameterized tests, test suites, and parallel testing, to ensure that existing functionality is not compromised. Simultaneously, we are incorporating the prototype development method. Initially, we create a simplified, interactive prototype showcasing the core functionalities and user interface of the system. We then review the requirements and use the prototype to validate the system's concept and user interface design. Once the concept validation is successful, we gradually add features, transforming the prototype into a usable system.

**1. .NET Core GUI( .Net 6.0) :**  .NET Core WinForms is a part of the .NET Core framework, specifically introduced in .NET 6.0, and it is used to build Windows desktop applications. It provides a rich set of libraries and features that simplify the development of Windows applications.

**2. User Interface (UI) Design :**

Windows Form App (WinForms): Windows Forms allows developers to customize the

appearance and behavior of applications in various ways.

Form:

It serves as a container for other controls.

Controls:

Button: Allows users to execute commands through clicks.

TextBox: Accepts user input as text fields.

Label: Displays text labels.

CheckBox: Permits users to select multiple options from a group.

RadioButton: Enables users to choose one option from a group.

ComboBox: Combines a text box and a list box, allowing users to select an item from the list or input a new value.

ListBox: Displays a list of items from which users can select one or more.

DataGridView: Presents tabular data, typically retrieved from a database source.

PictureBox: Used for displaying images.

DateTimePicker: Allows users to select dates and/or times via a dropdown calendar. Menu (ToolStripMenuItem): Provides a menu system, including top-level menus and context menus.

Container controls:

Panel: A simple container for grouping controls.

GroupBox: Displays a group box that can contain other controls.

Dialogs:

OpenFileDialog: Enables users to select files from the file system.

MessageBox: MessageBox can be used to show textual messages to users, such as information about application status, operation results, warning or error messages.

**3. Google api:** Google provides a powerful charting library and API, the fetch

function that loads JavaScript, and the google interface to implement the fold charts and pie charts, which enable the viewer to quickly understand the relative share of data, such as people role distribution data, categorical statistics, and so on. The

Google API also provides a number of customizable options that allow us to create

beautiful, customised pie charts that are consistent with your application or website. In our program, we enable features like zooming, hover effects, and click interactions to allow users to explore data more deeply and view detailed information. These

interactive elements enhance the user experience and make data analysis more intuitive and engaging.

**4. Azure function:** Azure Functions is event-driven and can respond to a variety of events and triggers, such as HTTP requests, etc.

**5. Random Library:** A library or module for generating random numbers, used in this system to generate memberid and productid and to verify the required random

numbers

**6. JavaScript fetch function:** an API for making network requests in Web applications. This system is used to communicate with the local server to obtain the data needed for pie charts and line charts

**7. Regex Module:** A regular expression is a pattern composed of a series of

characters and metacaracters, which is used to describe a specific pattern or format of a string. In this system, it is used to verify the E-mail format and telephone format entered by the user.

**8. Entity Framework (EF) :** In commodity purchase and management system, we use LINQ to create queries and tools such as SqlCommand to add parameters to ensure

that our queries are safe and Parameterized queries prevent SQL injection attacks.

we use EF to connect to databases containing information about products, orders,

and members. This includes Create, Read, Update, and Delete (CRUD) order, member, and product tables. For example, we can add new items, update order status, or

disable people. We also perform various database operations through parameterized queries. We apply the Database First Approach: creating models and connecting

them to DataGridView controls, TextBoxes, and Label controls based on an existing database.

**9. Microsoft Teams:** Used for team communication. Weekly progress meetings are held on Teams.

**10. Google Docs:** Create shared documents for collaborative editing.

**11. GitHub:** We create separate branches to develop features or fix bugs

independently. Then, we merge the changes from these branches into the main

branch through pull requests. Team members can review code written by others and provide feedback and suggestions.

**5. Flowchart**

