COMP9900 Project Report

Customer Relationship Management System

Team Name: 9900H11Bgogogo

Lab: 9900 H11B

[Scrum Master]

z5333461 Zheshan Zhang z5333461@ad.unsw.edu.au (Front-end)

[Group Member]

z5355922 Hongjia Ren z5355922@ad.unsw.edu.au (Front-end)

z5351815 Zhichao Xian z5351815@ad.unsw.edu.au (Front-end)

z5373811 Jinming Liu z5373811@ad.unsw.edu.au (Back-end)

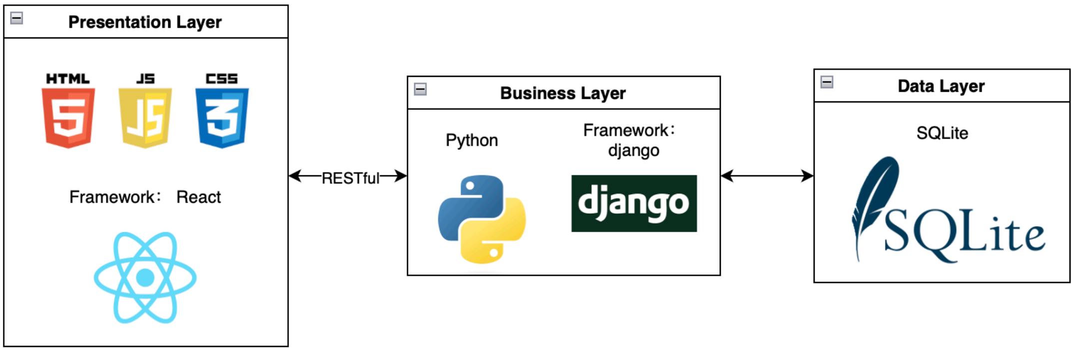
z5386953 Zhenxuan Liang z5386953@ad.unsw.edu.au (Back-end)

Submission date: 4 August 2023

Catalogue

1. Overview

1.1 System Architecture



Picture 1.1 system Architecture

1.1.1 Presentation Layer

The presentation layer deals with data transfer and presentation, and is the layer used and interacted with by the users of the system. For the presentation layer, users can directly use a series of functions of the system. This system uses the front-end framework react technology to display the pages of the system website, in which the style of page display is determined by the system designer and developer, in order to let users have a good user experience. The reason for choosing React framework technology is that componentized development method can simplify complex UI process, can help CRM system development, and improve development efficiency. The advantage of React framework technology is that it has virtual DOM and high performance, which is conducive to the maintenance and extension of user interface. When developing CRM systems, these features bring us a better development experience and higher performance.

1.1.2 Business Layer

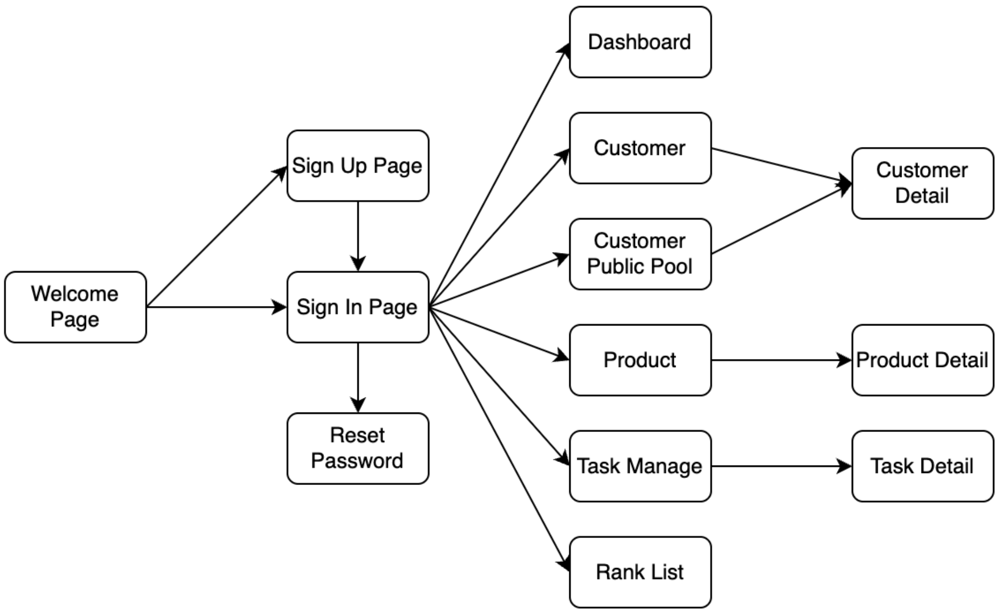
This layer controls the Django Restful API processing, it follows separating concerns and ensuring modularity. The request from Presentation layer will be routed to specifical module through the URL pattern. Then, the corresponding query will be sent to data layer. Serializer pattern onvert instances into serialized formats like JSON, facilitating data transmission and handling validation. The processed data based on the query will be sent back to the previous layer as return.

1.1.3 Data Layer

The data layer based on SQLite3 and Django's ORM (Object-Relation Mapping) in the back-end to interact with. The ORM simplifies data access by offering an intuitive interface for creating, retrieving, updating, and deleting records. ORM will be used to implement some functions easily like retrieving customer information, updating customer details, and querying the database. Through ORM, the layer efficiently retrieves the database and returns the results to the business layer.

1.2 Page Structure

Welcome Page: A brief introduction of our system.
Sign Up Page: Account registration for new manager/staff.
Sign In Page: Sign in for an existing account.
Dashboard: Data statistic of customers and products.
Customer Page: Shows your customer (for staff) or all customers with principle (for manager).
Customer Public Pool: Shows all the customer without principle.
Customer Detail Page: Shows the detail of designated customer.
Product Page: Shows all the products of the company.
Product Detail Page: Shows the detail of designated product.
Task Manage Page: Shows your tasks (for staff) or all the tasks (for manager).
Task Detail Page: Shows the detail of the designated task.
Rank List Page: Shows the personal detail and the rank list of top 10 employees.
Reset Password Page: Reset the account's password.



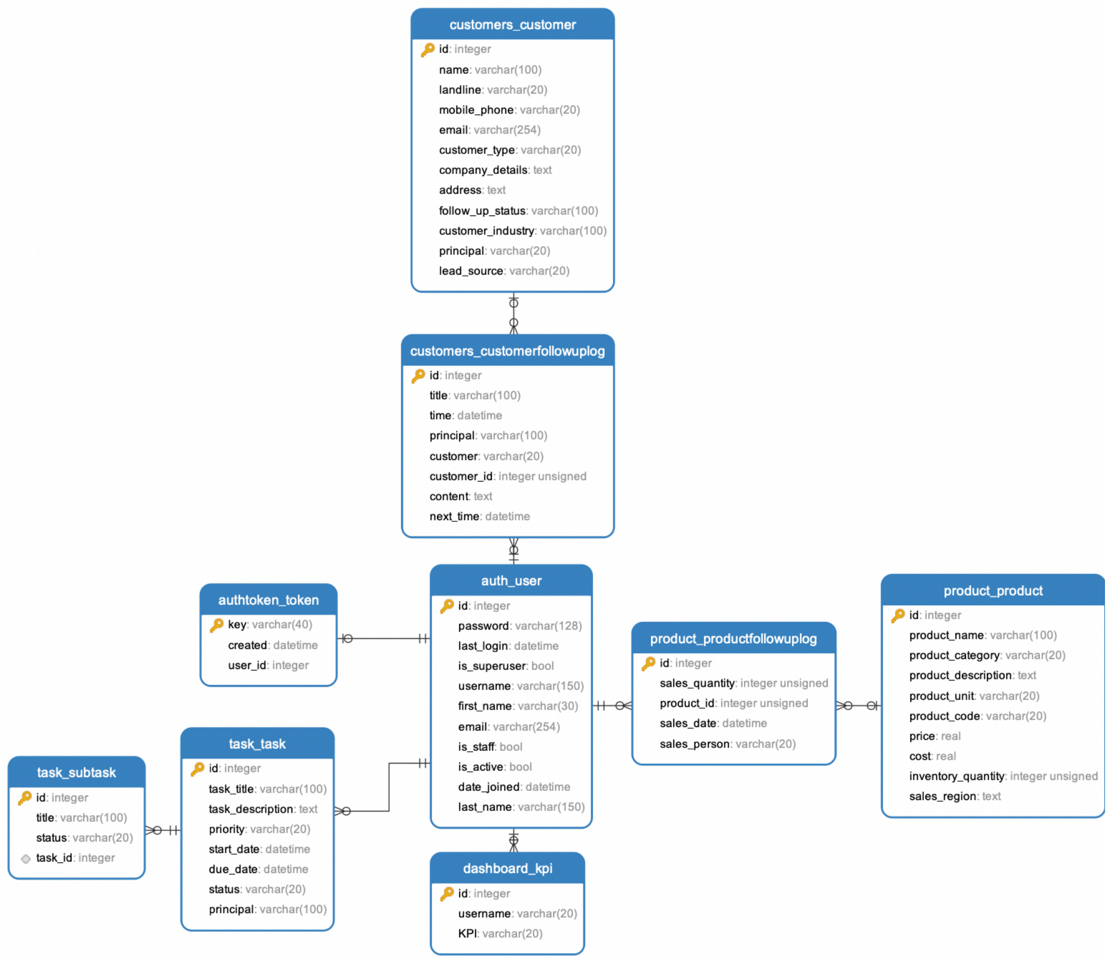
Picture 1.2 Page Structure

1.3 Database Design

The CRM system database contains 9 tables. The implement of user authentication system needs auth_user and authtoken_token tables. The auth_user table store the manager/staff personal information like name and role when they register, and the authtoken_token table contains the token of user which is used for authentication instead of password.

The tables customers_customer and customers_customerfollowuplog are used for customer module. When the staff/manager add a new customer, there would be a new record in customers_customer and each time a customer related to the customers_customer is followed up, there would be a new record in customers_customerfollowuplog.

For the task module, the task_task table store the main task of different staff and each task can have several sub tasks in task_subtask table. What's more, sales records relate staff/manager and specific item, is stored in product_productfollowuplog table. Finally, the dashboard_kpi table stores the staff/manager KPI information.



Picture 1.3 Database Structure

1.4 Project Objectives

Project Objective 1: User Authentication System
Create a secure and user-friendly authentication system where managers and staff can register, login, and log out using their username and password credentials.

Related Project Objectives	Project Objective 1
User stories	As a staff, I want to register a staff account so that I can log in to the system. As a manager, I want to register a manager account for log in to the system and manage work. As a staff or manager, I want to be able to log out the system so that I keep my information safe. As a staff or manager, I want to have ability to log into the system so that I can access its features. As a staff or manager, I want to reset my password in case I forget it.

Table 1.4.1 PO1

Project Objective 2: Sales Dashboard Development
Develop a highly visual and engaging sales dashboard that presents various charts and indicators using a customized interface. This will allow managers and staff to gain insights into sales performance.

Related Project Objectives	Project Objective 2
User stories	As a staff, I want to see my own business sales sheet so I can see my sales. As a staff, I want to see the sales chart of the company's products so that I can learn about the company. As a manager I want to see employee performance charts so you can see if your employees are up to par. As a manager I want to see the inventory of the company's products so that you can understand the popularity of the products. As a stuff, I want to see the total sales table to understand the company's turnover.

Table 1.4.2 PO2

Project Objective 3: Customer Contact Information Management
Enable managers and staff to enter, view, and manage the contact information of customers effectively within the CRM system.

Related Project Objectives	Project Objective 3
User stories	As a staff/manager member, I want to be able to use new customer

	information to build customer profiles or delete the invalid customer information. As a staff/manager member, I hope that I can search specific customer information so that I am able to monitor the business-specific matric. As a staff member, I hope that I can update customer details like phone number. As a staff, I want to view the interaction history of specific customer to keep track of customer journeys. As a staff member, I want to be able to transfer customer information that I no longer follow to the company's public customer information pool.

Table 1.4.3 PO3

Project Objective 4: Task Management System

Develop a system that allows managers to create, assign tasks to personnel, with the ability to specify task priority levels, deadlines, and descriptions. Empower staff to view tasks assigned to them and to update task status to 'in progress' and 'complete' for effective task management and completion tracking.

Related Project Objectives	Project Objective 4
User stories	As a manager, I want to create and assign tasks to staff. As a manager, I want to set detail information of the task, which include name, details, priority level, deadline and status. As a manager, I want all tasks can be highly visualized so that I can manage workflows. As a staff, I want all tasks can be highly visualized so that I can manage my work. As a staff, I want update tasks status.

Table 1.4.4 PO4

Project Objective 5: Inventory Database for Products

Design a feature allowing managers and staff to add company products into an inventory database to track product performance with customers and identify lost sales opportunities.

Related Project Objectives	Project Objective 5
User stories	As a staff, I want to add new products to the inventory database to keep track of product inventory. As a manager, I want to search the inventory status of all products in order to make inventory management decisions. As a manager, I want to change the product inventory information for better statistics. As a staff, I want to look at product inventory to see how individual products are selling.

	As a manager I want to search for individual products in order to understand individual product turnover

Table 1.4.5 PO5

Project Objective 6: Advanced Search and Filter Capabilities

Provide managers and staff the ability to search for a specific product or customer and filter customers or products based on different criteria for improved efficiency and data management.

Related Project Objectives	Project Objective 6
User stories	As a staff, I want to search for a specific customer that I own, so that I can quickly view the relevant information of the customer, such as the contact information of the customer, which is conducive to improving the efficiency of my contact with customers. As a staff, I want to search for a specific product so that I can quickly see information about the product, such as the price of the product, so that I can quickly quote the customer. As a staff, I want to filter the customer list according to different criteria, so that I can generate the customer list according to the conditions, I can get a set of information, which is conducive to my batch management of multiple customer information. As a staff, I want to filter the list of products according to different criteria so that I can more easily view information about a group of products, which helps me compare different products to customers. As a manager, I want to search for a specific customer in all areas so that I can quickly view relevant information about the customer, such as the contact information of the customer, which is conducive to improving my staff in contacting the customer, and to contacting the customer urgently as a manager.

Table 1.4.6 PO6

Project Objective 7: Customer Public Pool

Development of a Shared Customer Resource Pool. This Epic aims to develop a shared customer resource pool that can be accessed by both staff and managers.

Related Project Objectives	Project Objective 7
User stories	As a staff, I want additional customer resources from the system so that I can acquire other customers. As a staff, I want additional customer resources from the system for the company, so that other employees can acquire other customers. As a manager, I want the company's customer resources will not be lost, and customer resources will circulate among all employees to improve the success rate of customers.

	As a manager, I want the company to have a common customer resource pool that allows me to allocate different customer resources. As a manager, I want to manage customer common resources, which is conducive to improving the efficiency of the company's customer resources.

Table 1.4.7 PO7

Project Objective 8: Gamification Elements

The objective of this project is to introduce and implement gamification elements, including points, badges, and leaderboards, in the CRM system to enhance personnel's work performance.

Related Project Objectives	Project Objective 8
User stories	As an employee, I want to earn points for completing work tasks, so that I can be motivated to work harder. As an employee, I want to receive badges when I achieve specific targets or complete significant tasks, to display my achievements and motivate me to improve further. As an employee, I want to see a leaderboard showing the points and badges of all employees, so I can understand how my performance compares with others in the team. As an employee, I want to be able to redeem rewards or benefits with the points I earn, so I can see tangible returns for my efforts. As an employee, I want to view my points and badges on my personal page, so I can understand my progress and how far I am from the next target.

Table 1.4.8 PO8

2. Functionality Detail

2.1 Sign in Page

Related to: Project Objective 1

In the sign in interface, the main purpose is for user sign in. On this interface, there are input fields for the username and password, as well as a submit button. All possible conditions are listed in the table below:

Condition	**Result**
Username or password is correct	Sign in successfully, jump to Dashboard Page
Username or password is not correct	There will have a red warning appear “Please check the username or password”

The API for Sign in Page shows below:

API name	URL
Sign in	POST /localhost:8000/api/user/login/

2.2 Sign up Page

Related to: Project Objective 1

In the Sign-up interface, the main purpose is for user registration. On this interface, there are fields for the username, password, email address, first name, last name, and an option to register as an administrator. All possible conditions are listed in the table below:

Condition	Result
Enter all information correctly	Sign up successfully, jump to Sign in Page
The password and second password are not match	There will have a red warning appear “Password do not match”

The API for Sign up Page shows below:

API name	URL
Sign up	POST /localhost:8000/api/user/register/

2.3 Dashboard Page

Related to: Project Objective 2

In the dashboard page, you can see customer and sales statistics, as well as individual performance indicators. Provides user-friendly information presentation through high data visualization like bar graphs, line charts, pie charts. All possible scenarios can be seen in the table below:

Condition	Result
Click line button	Shows the line chart
Click pie button	Shows the pie chart
Choose different customer types	Shows the bar chart of different customer type statistics
Click set KPI button and input target	Record KPI data and count the degree of completion

The API for Dashboard Page shows below:

API name	URL
Dashboard	GET /localhost:8000/api/dashboard/customerstatistic/

2.4 Customer Page

Related to: Project Objective 3&6

In the customer page, you can see the basic information of the customers, with buttons for category search and adding new customers. Users can see the customer’s detail by clicking specific customer name. What’s more, the selected customers can be deleted or moved to the customer public pool. All possible scenarios can be seen in the table below:

Condition	Result

Click search button	Shows all customers match the requirement you have chosen
Click the add customer button	Add a new customer to the database
Choose customers and click the ‘transfer to public pool’	The customers were chosen will be transferred to the customer public pool
Click customer name	Shows the customer detail

The API for customer page shows below:

API name	URL
Customers	GET /localhost:8000/api/customer/customers/
Add customer	POST: /localhost:8000/api/customer/customers/
Transfer customers	POST /localhost:8000/api/publicpool/
Customer detail	GET: /localhost:8000/api/customer/detail/{customer_id}/

2.5 Customer Detail Page

Related to: Project Objective 6

In the customer details page, you can see the detailed information of the current customer, with buttons on the page for editing, deleting, and adding customer follow-up log entries. All possible scenarios can be seen in the table below:

Condition	Result
Click Edit Button	Edit this customer’s detail
Click Delete Button	Delete this customer
Click ‘New Timeline’ button	Add a new customer follow-up log entry

The API for Customer Detail Page shows below:

API name	URL
Edit customer	PUT: /localhost:8000/api/customer/customers/update/
Delete customer	DELETE:/localhost:8000/api/customer/customers/{customer_id}/
Update customer	PUT: /localhost:8000/api/customer/customers/update/
Follow-up log	POST: /localhost:8000/api/customer/follow-up/

2.6 Customer Public-Pool Page

Related to: Project Objective 6&7

In the customer pool page, you can see customers placed in the pool (customers without a person in charge). In this interface, you can bulk claim selected customers, bulk assign selected customers, bulk delete selected customers, and categorize and search for customers. All possible conditions can be seen in the table below:

Condition	Result

Click Search Button	Shows all customers match the requirement you have chosen
Click Retrieval Button	Change the customer’s principal to you
Click Allocation Button	Allocate the customers to the staff you enter (only manager can do)
Click Delete Button	Delete the customers in public-pool (only manager can do)

The API for Customer Public-Pool Page shows below:

API name	URL
Search customer	GET: /localhost:8000/api/publicpool/
Allocation	PUT: /localhost:8000/api/publicpool/allocation
Retrieval	PUT: /localhost:8000/api/publicpool/pickup
Delete	DELETE: /localhost:8000/api/publicpool/

2.7 Product Page

Related to: Project Objective 5&6

In the product page, you can see the basic information of the company's products, and you can search for products based on product name, product price, and add products. Clicking on the product's ID will show you the detailed information of the current product. In the detailed information interface, manager can delete products. Both manager and staff can add a sales record for the product which is used to count individual sales and calculate the popularity of a product. All possible conditions can be seen in the table below:

Condition	Result
Click search Button	Shows all Products match the requirement you have chosen
Click Add Product Button	Add a new product to database
Click Product ID	Shows the detail information of this product
Click Add Record button	Add a new sales order to the sales records

The API for Product Page shows below:

API name	URL
Search Product	GET: /localhost:8000/api/products/?product_name=iPhone13
Add Product	POST: /localhost:8000/api/products
Check Product	GET: /localhost:8000/api/products/{product_id}
Add Record	POST: /localhost:8000/api/products/{product_id}

2.8 Task Page

Related to: Project Objective 4&6

In the task management page, you can filter tasks based on the task's name, priority, deadline and so on. By clicking the "Add Task" button, manager role can create and assign a new task to staff or himself. Clicking on the task's title will show you the task's details and completion status (including the completion status of subtasks). You can modify the task status and delete tasks on this interface. All possible scenarios can be seen in the table below:

Condition	Result
Click Search/Filter Button	Shows all Tasks match the requirement you have chosen
Click Add Task Button	Add a new task and allocate it to a staff
Click Task Title	Shows the detail information and the status of this task
Click Delete Task button	Delete this Task
Click Status button	Change the Tasks/Sub Tasks status

The API for Task Page shows below:

API name	URL
Search Task	GET: /localhost:8000/api/tasks/?param1=value1
Add Task	POST: /localhost:8000/api/tasks/
Check Task	GET: /localhost:8000/api/tasks/{task_id}/
Delete Task	DELETE: /localhost:8000/api/tasks/{task_id}/
Change Status	PUT: /localhost:8000/api/tasks/{task_id}/

2.9 Rank List Page

Related to: Project Objective 8
In the leaderboard interface, you can see your sales ranking, the points earned, and the rewards that can be obtained upon reaching a certain number of points, thereby motivating employees.

The API for Task Page shows below:

API name	URL
Get Rank	GET: /localhost:8000/api/rank/
Get personal info	GET: /localhost:8000/api/rank/myself/

3. Technologies Description

3.1 Third-party Functionalities

3.1.1 Front-End

React
React technology is a very popular front-end framework technology, developed by Facebook and used to build user interfaces. The reason we chose to use the react framework technology, because

the react technology is more mature now, we can get a lot from the Internet the react framework of technical support. Another advantage of the react framework is that a component development approach can simplify complex ui processes. React also has the benefits of virtual DOM and high performance, so our user interface is maintainable and extensible. Another huge advantage of react as a front-end framework is that it is very friendly to entry-level front-end developers.

Material-UI

Based on Google's Design language "Material Design", with a modern graphic design style, Material-UI was created in 2014 by Olivier Tassinari and the Mui-Toolbelt team and released at Google the same year. Material-UI has many advantages, its design is very beautiful, and the style is unified, highly consistent and beautiful. The Material-UI has a large library of components, such as buttons, tables, navigation bars, etc., which can help developers build user interfaces efficiently and quickly. The components of Material-UI can be laid out in a responsive manner, and the most important thing is that they can adapt to different screens and different kinds of devices, which has an excellent user experience. It is also a feature of Material-UI that developers can easily customize the style and style of the components they need. Material-UI has a mature and strong community, is very friendly to developers, can easily find relevant information, low cost of learning and use, is very conducive to the development of beginners.

3.1.2 Back-End

Django Restful Framework (DRF)

DRF is a Python web develop framework for restful API. It is a powerful and flexible tookit for building Web APIs using the Django framework. DRF provides useful tool set which include authentication, ORM data source and other utilities.

Google Gmail SMTP server

SMTP server is a transit server for the mail sending function in the system.

3.2 Challenges and Difficulties

1. Challenge: Implement "Reset password".

In order to prevent the user from forgetting the password, it is necessary to confirm the user's identity information through means other than the account password and provide the function of changing the password for the authenticated user. A feasible method is a security question, but there is a possibility that the user may forget the answer to the security question. Another method is needed, which is to register the email address and change the password.

Solution: By import the Django. Mail library, configure the Google SMTP server as a relay server to send emails. Users can complete the password modification through the password modification link in the email.

2. Challenge: Permission system

Managers and staffs have different permissions to complete corresponding operations. To verify whether the request has permission to perform the operation is a problem that needs to be solved.
Soltuion: Design a token authentication system. Use token to identify the role of the user.

3. Challenge: CORS permission problems

The team in the process of development, appeared when a web application tries to request resources from different domains, the browser will refuse to cross-domain request. Research by team members found that this was a Cors permission issue, including cases where the request was denied, and the browser got an error message indicating that the request was denied.
Solution: configure the CORS header.

4. Challenge: Usability and Responsiveness of the User Interface (UI)

In CRM systems, the usability and responsiveness of the User Interface (UI) are vital for the overall user experience. Since CRM systems often involve substantial data and multiple functionalities, designing an intuitive, easy-to-use, and responsive UI is very challenging.
Solution: Collaborate with end-users for requirement analysis, create prototypes and wireframes to ensure that the system is navigable and meets user needs. And we use React to enhance page responsiveness and provide rich user interaction experiences.

4. Installation and Prerequisites

4.1 Run on the local environment

4.1.1. Download and Initialization

1.1 Clone the code on your local environment:
    Git clone:
    https://github.com/unsw-cse-comp3900-9900-23T2/capstone-project-9900h11bgogogo.git
1.2 Install python3 in your local environment.
1.3 Install Node.js 14.17.0 in your local environment:
1. Go to front-end director:
    Run: cd 9900Project/front-end
2. Download the nvm installation script using curl:
    Run: curl -o- https://raw.githubusercontent.com/nvm-sh/nvm/v0.39.3/install.sh | bash
3. Close and reopen your terminal window or run the following command to apply the changes
    Run: source ~/.bashrc
    if no such file:
    Run: touch ~/.bashrc

then apply again:
Run: source ~/.bashrc
4. Verify that nvm has been installed correctly by running the following command
Run: nvm --version
This command should return the current version of nvm installed on your system, which should be v0.39.3
5. Install version 14.17.0
Run: nvm install 14.17.0
6. Verify that Node.js has been installed correctly by running the following command:
Run: node -v
This command should return the version of Node.js that you just installed (14.17.0).If not, Please reinstall version 14.17.0

4.1.2. Run Back-end server

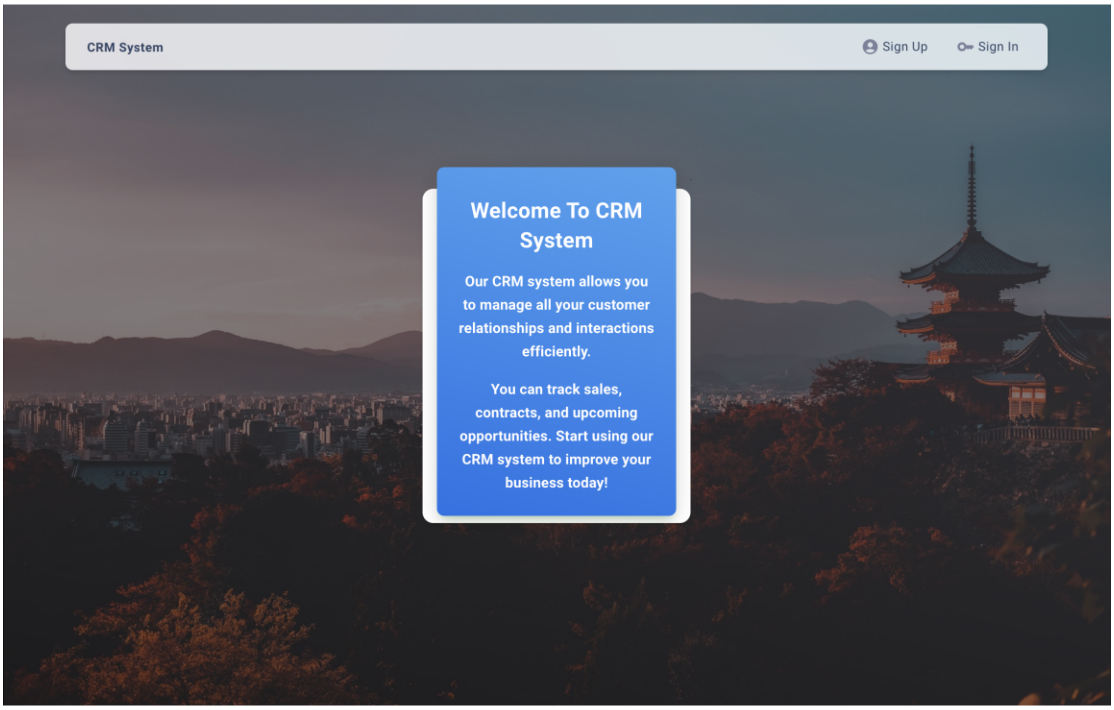
1. Go to ‘back_end’ file director:
Run: cd 9900Project/back_end
2. Run: pip install -r requirements.txt
3. Go to ‘crm’ director:
Run: cd 9900Project/back_end/crm
4. Run: python3 manage.py runserver

4.1.3 Run Front-end website

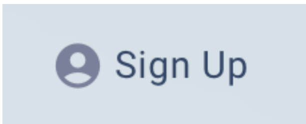
Before using our services like login or register, you should open and run the backend first because database connection is required.
1. Go to ‘build’ file director:
Run: cd 9900Project/front_end/build
2. Run: npm install -g serve
3. Run: serve -s

5. User Documentation

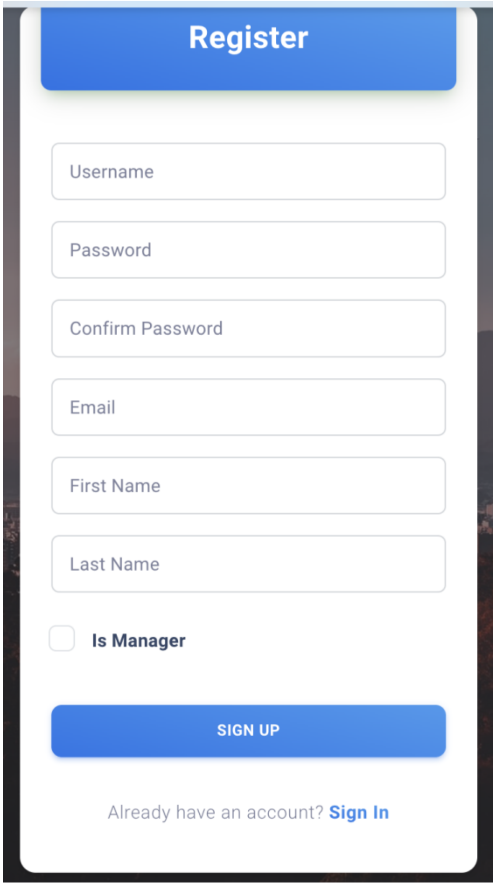
1. The following is the Landing Page of our website. There is a brief system introduction and welcome message.
The address is ‘http://localhost:3000/’.



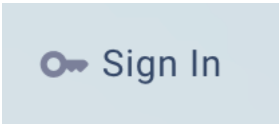
2. User can register by clicking the ‘sign up’ button to sign up an account.
The address is ‘http://localhost:3000/authentication/sign-up’.



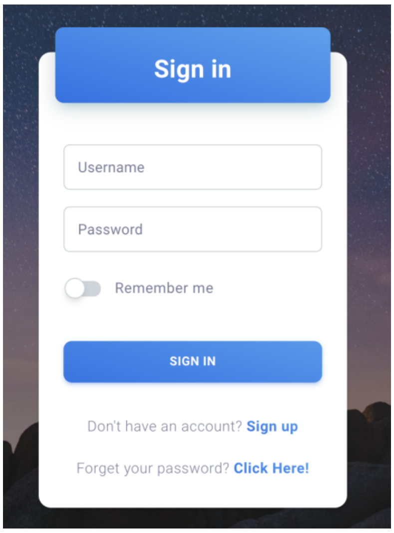
3. If the user already has an account, please jump to step 4. The following shows the register form. After filling out the information in the form and clicking the submit button, you can complete the registration.



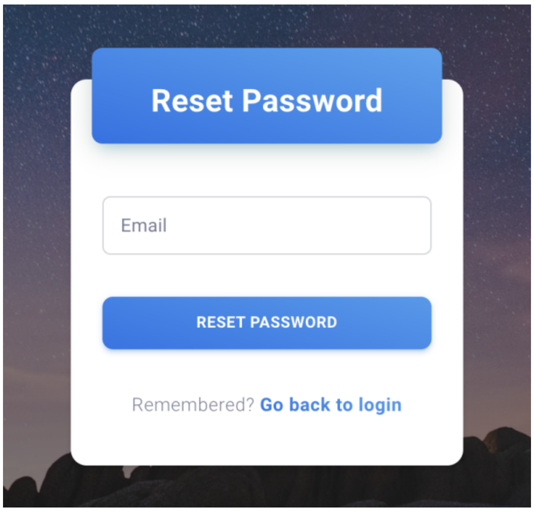
4. If user has an account, click ‘sign in’ button to log into the system.
The address is ‘http://localhost:3000/authentication/sign-in’.



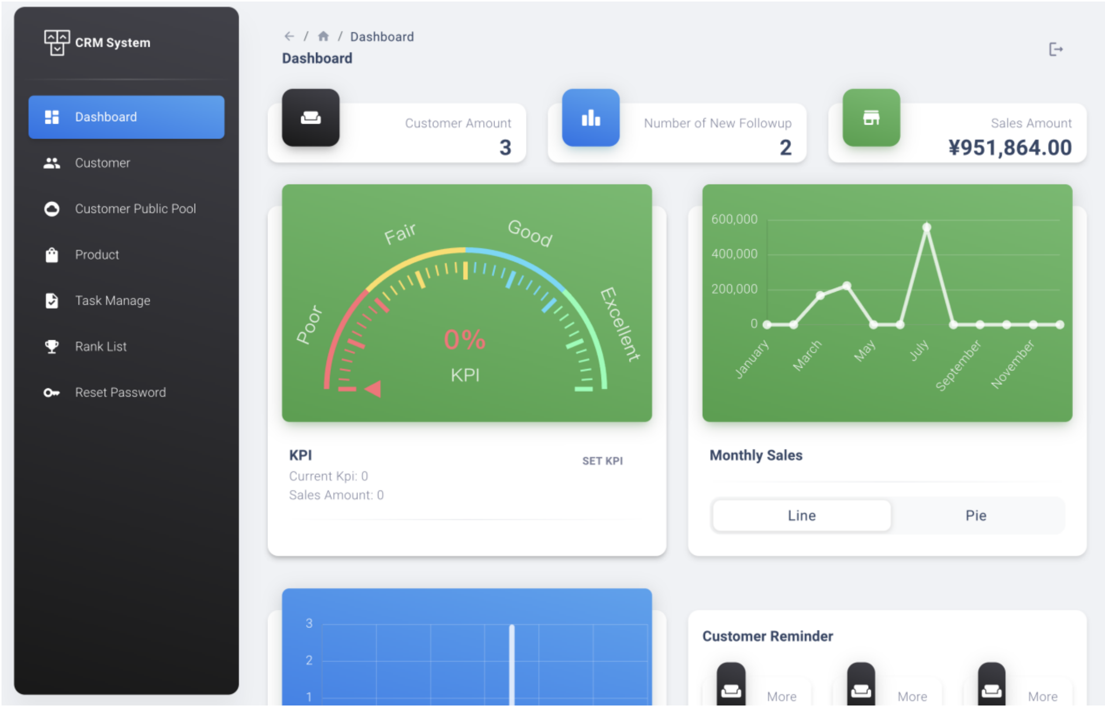
5. The following shows the sign in form. Once you have filled out the information in the login form and clicked submit, you can enter the dashboard interface. If you have forgotten your password, you can click on 'click here' on this interface to reset your password via email.



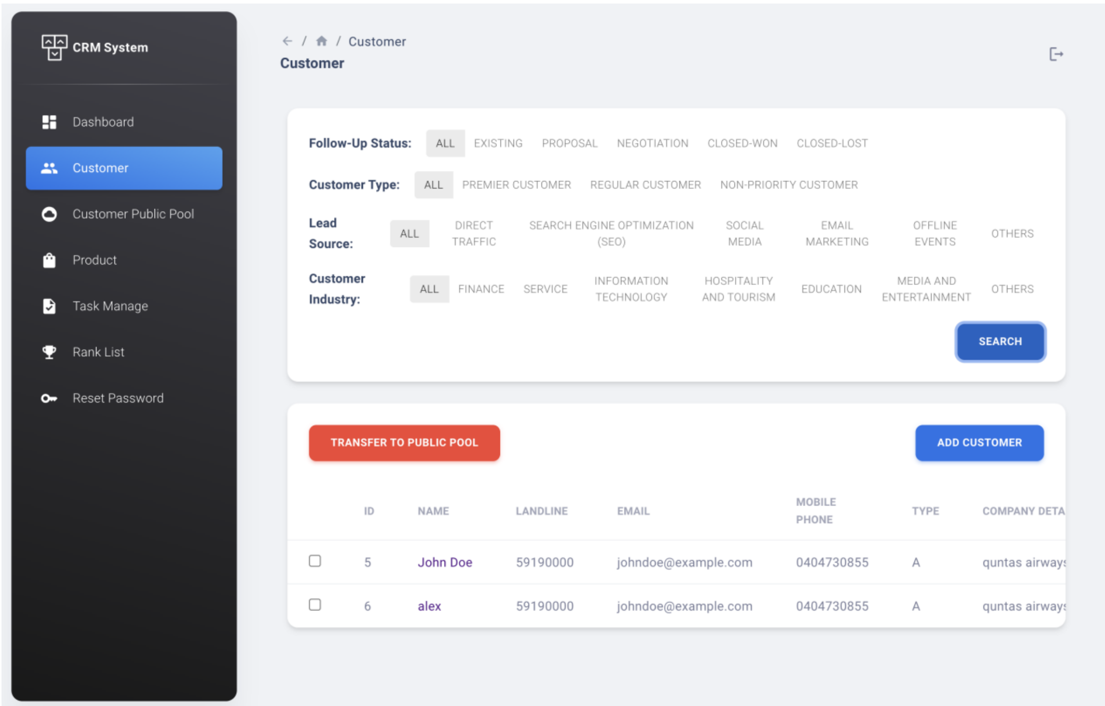
6. If you want to reset your password by email, you will see the following page. You will receive an reset password email after you submit your email address(the email address you used for registration).
The address is ‘http://localhost:3000/reset-email’



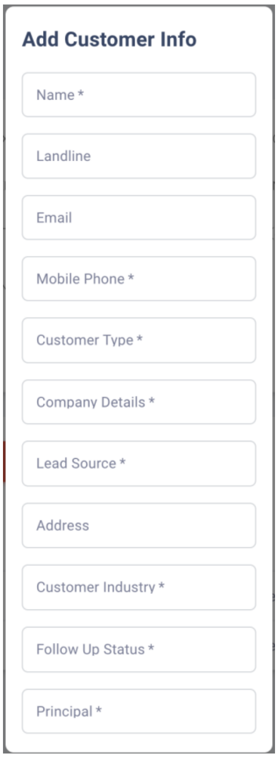
7. After you sign in successfully, you will see the following page -- the Dashboard. In the dashboard interface, you can see many basic details. If you are a manager, you can view the entire company's customer information and sales information, all presented to you in the form of charts. If you are staff, then you can only see the customer information and sales information that belong to you. Additionally, logged-in users can also view the status of their own KPI completion.
The Dashboard address is ‘http://localhost:3000/dashboard’.



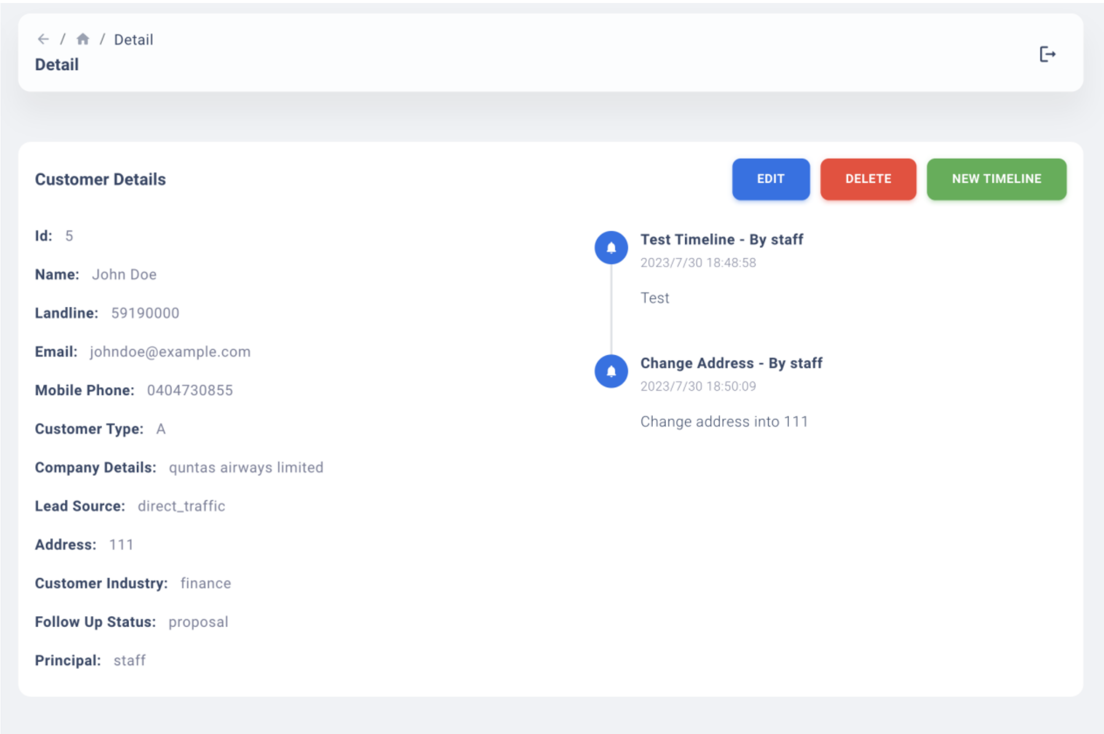
8. The Customer Page shows the basic information of customer. In the customer interface, there are a total of three buttons: ‘search’, ‘add customer’, and ‘transfer to pool’. The ‘search’ button allows you to filter customers that meet your criteria based on your selection, the ‘add customer’ button is used to add customers (jump to step 9), and the ‘transfer to pool’ button can place unwanted customers into the customer public pool (jump to step 13). You can also click on a customer's name to enter the customer details interface (jump to step 10).
The address is ‘http://localhost:3000/customer’.



9. After you fill the following form and submit, you can create a new customer.



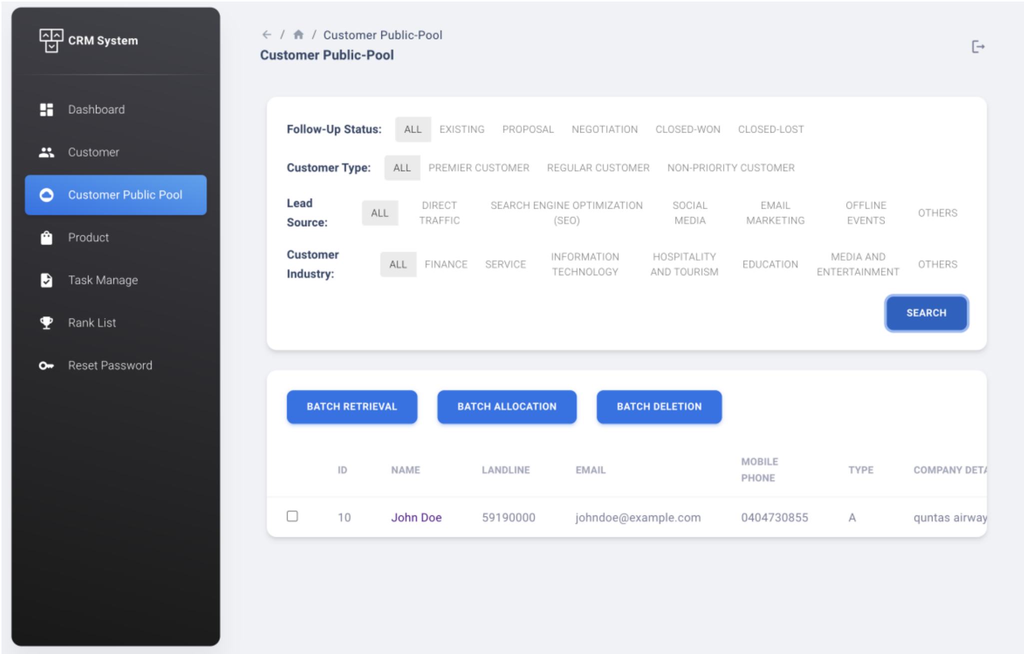
10. You can see the customer detail after you click the customer's name. In this interface, you can not only display customers' basic information but also delete customers, edit customer information, and add customer follow-up logs.
The address is 'http://localhost:3000/detail?id={customer_id}'.



11. The Customer Public Pool Page shows the basic customer information of customer public pool. Similar to the customer interface, you can also click the search button to filter different types of

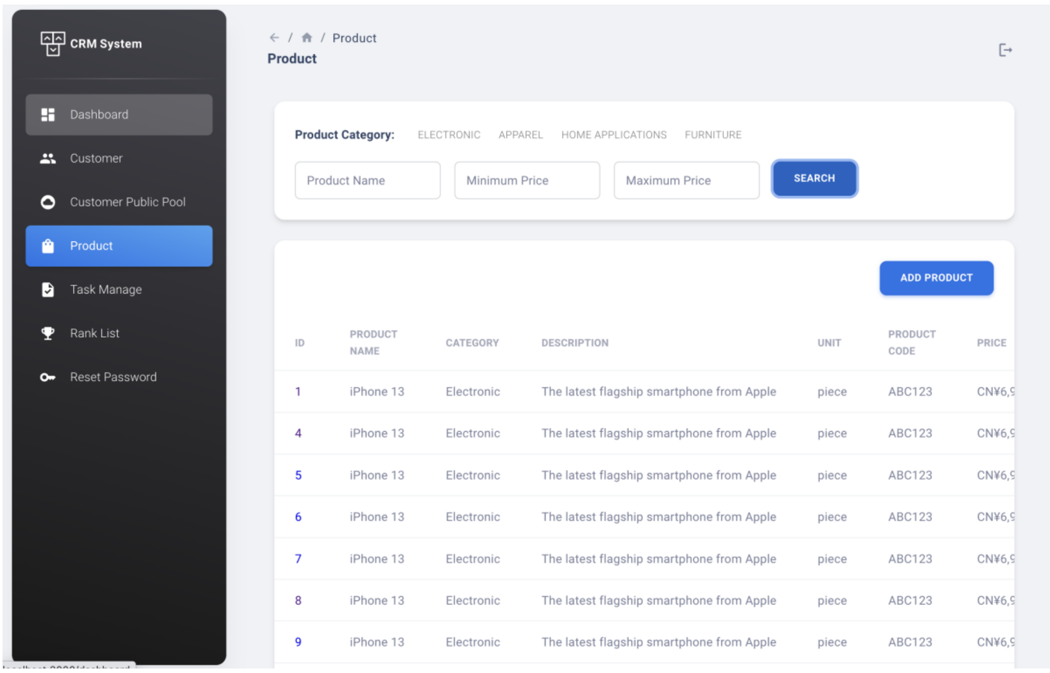
customers. If you are a manager, you can also bulk delete customers from the customer pool and bulk assign customers to the staff you wish to assign. If you are staff, then you can obtain the customers you want from the customer pool into your own customer list. Additionally, if you click on a customer's name in the customer pool, you can also enter the customer details interface (jump to step 10).

The address is ‘http://localhost:3000/customer-public-pool’.

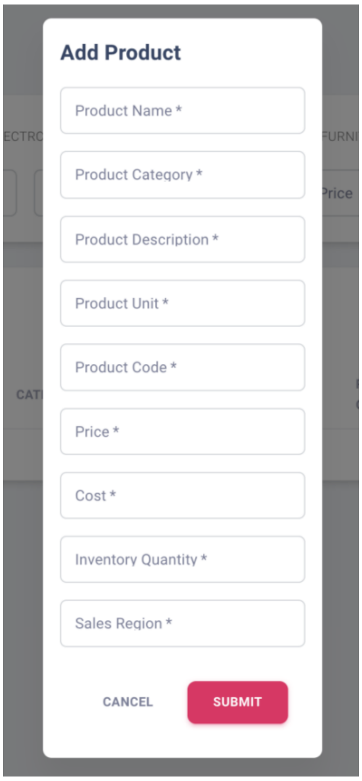


12. In the product interface, you can see the basic information of the product and can perform a fuzzy search based on the product's type, name, and price range. Clicking on the product's ID will take you to the product's detail interface (step 14). If you are a manager, you can click the 'add product' button to add a new product (step 13).

The address is ‘http://localhost:3000/product’.

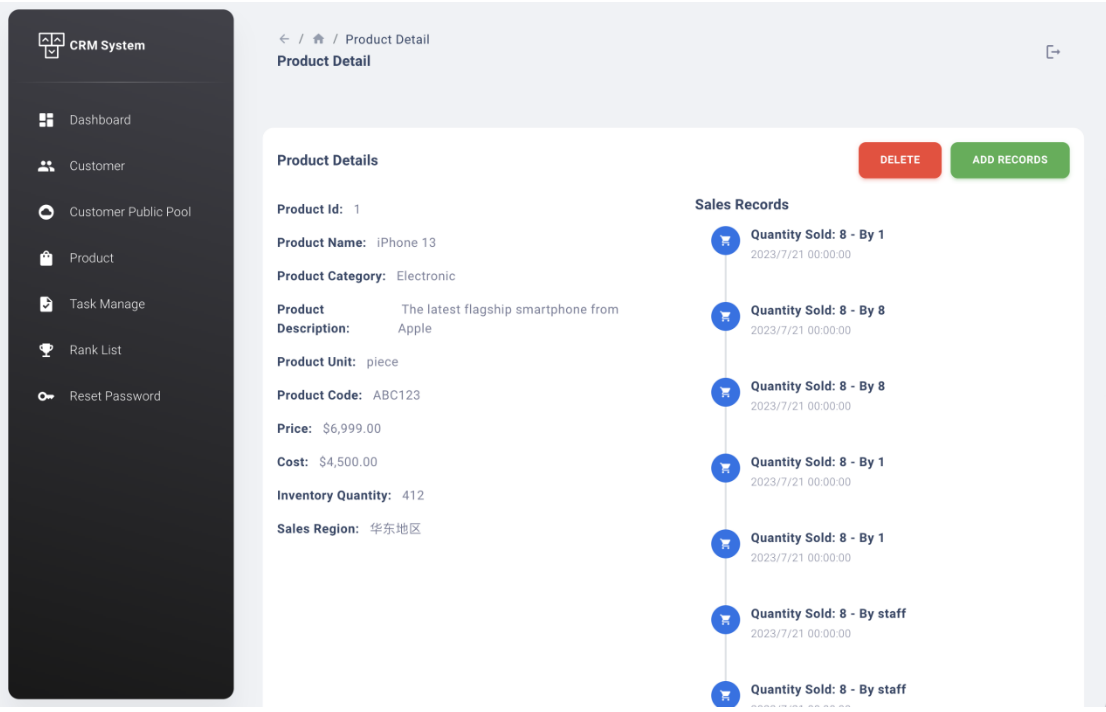


13. After completing the following form, you can add a new product.

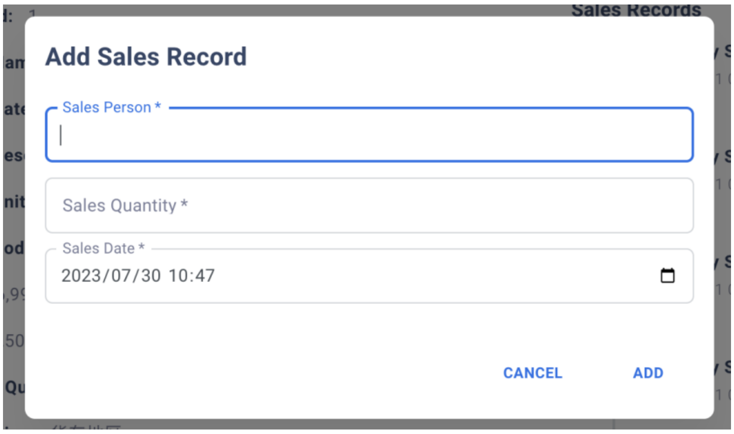


14. In the product's detail interface, you can see detailed information about the product. To the right of the product's information, you can see the product's sales records. By clicking the 'add record' button, you can add a new sales record (step 14), representing that you have successfully sold an order of the current product.

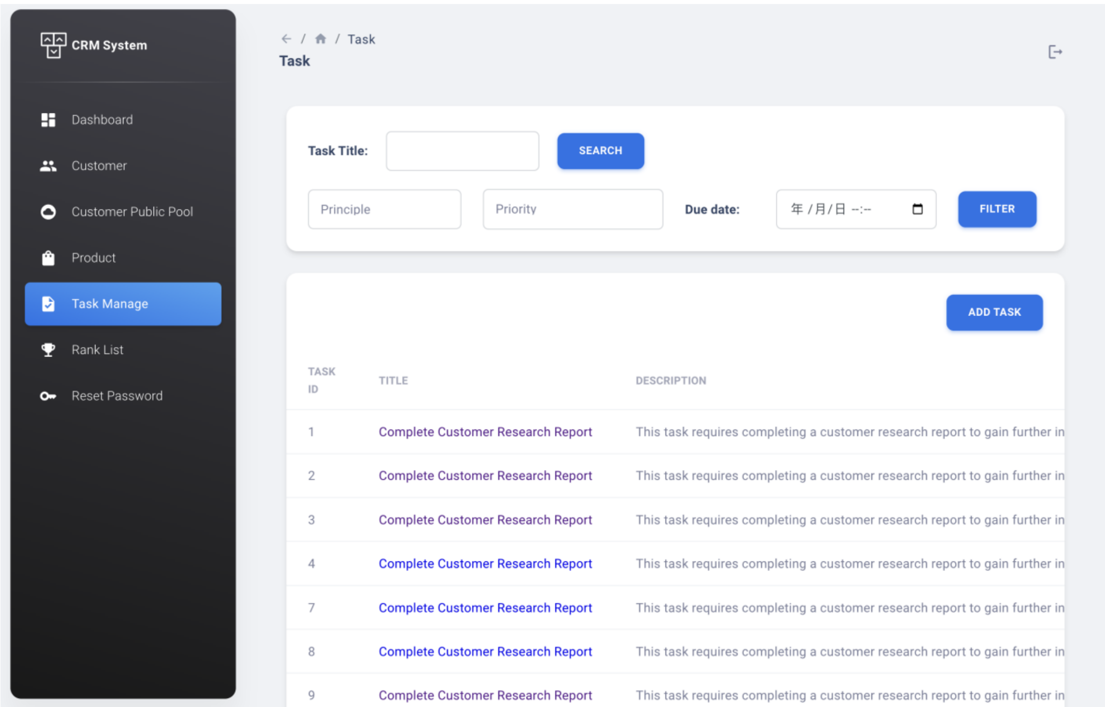
The address is 'http://localhost:3000/product-detail?id={product_id}'.



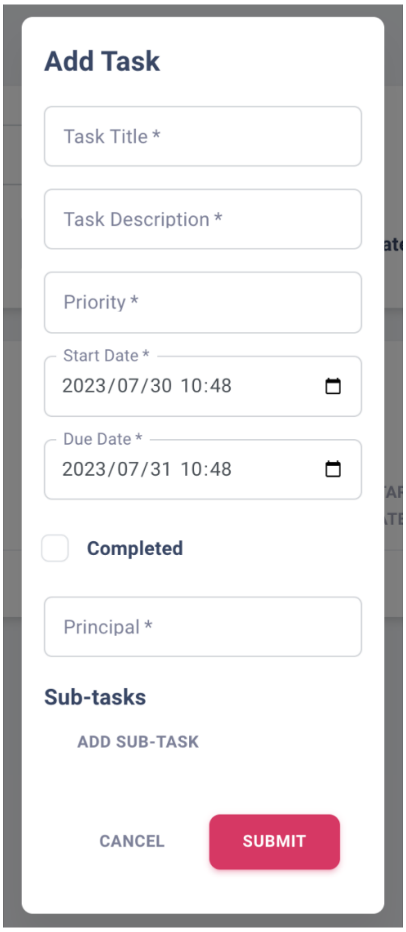
15. After completing the following form below and clicking submit, you can add a sales record.



16. If you are a manager, then in the task management interface you can see all the task information, and you can assign a new task to staff by clicking the 'add task' button (step 17). If you are staff, you can only see the tasks assigned to you. Everyone can perform a fuzzy search for tasks based on the task's title, priority, and deadline. You can also click on the task's title to enter the task's detail interface (step 18).
The address is ‘http://localhost:3000/task’.

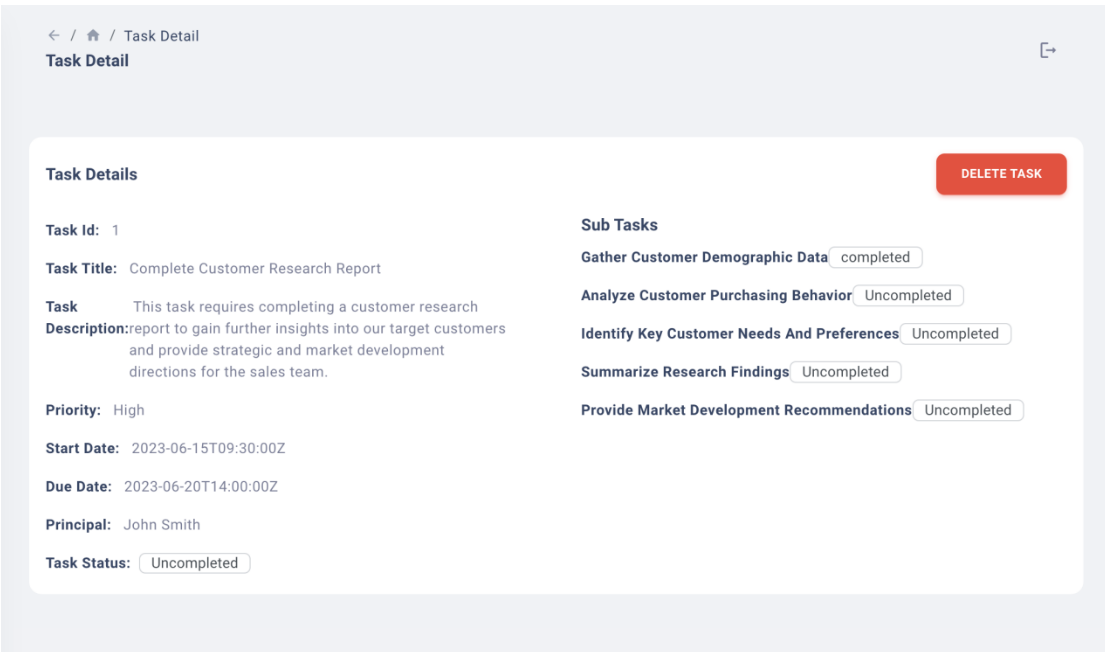


17. After completing the following form below and clicking submit, you can add a new task.

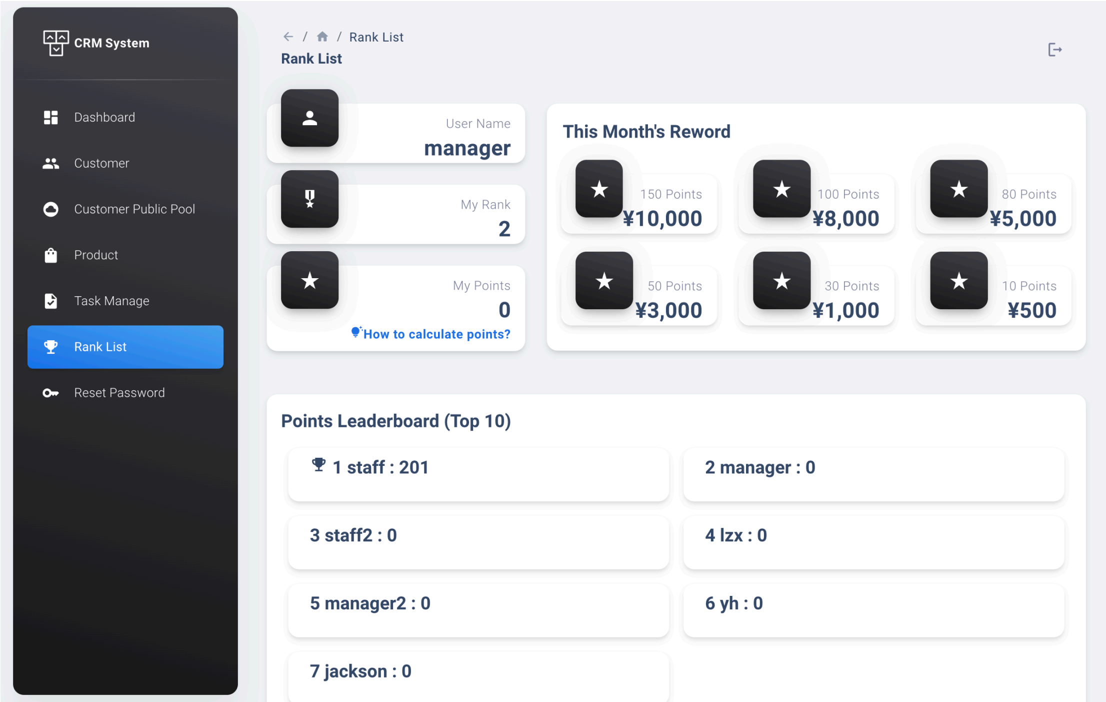


18.In the task detail interface, you can see the information of the current task and subtask, and you can also change the completion status of the task.

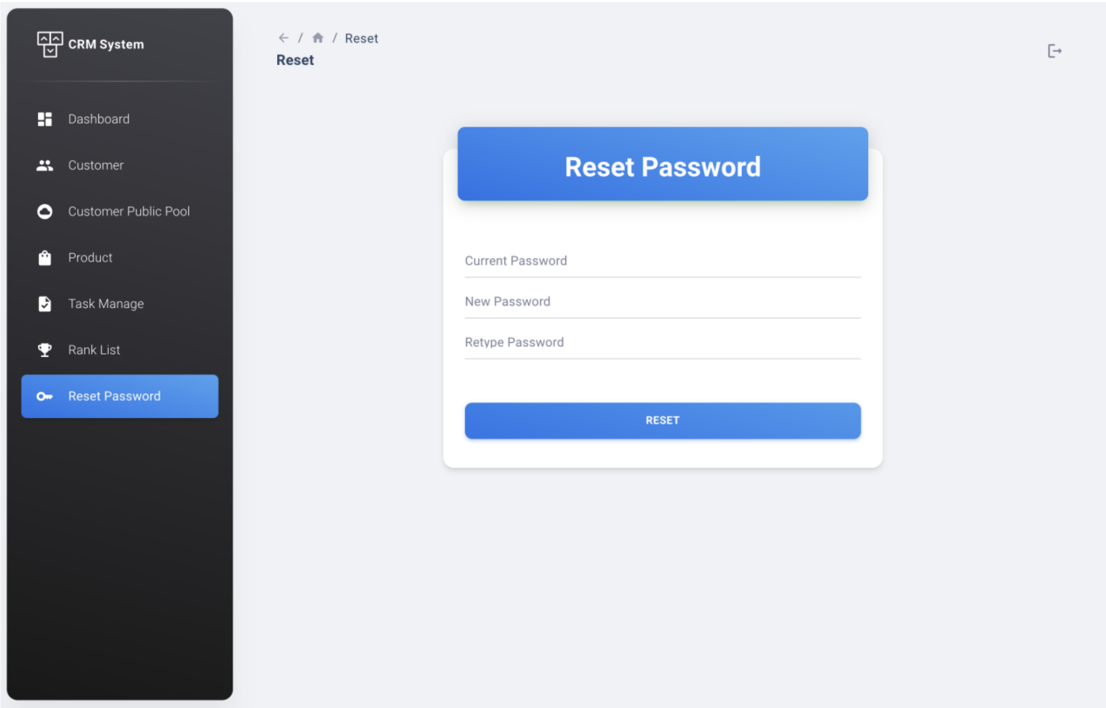
The address is ‘http://localhost:3000/task-detail?id={task_id}’.



19. In the Ranklist interface, you can see your username, points, and point ranking. On this interface, you can also see the current month's point rewards and the points leaderboard for all employees of the month (top ten).
The address is ‘http://localhost:3000/rank-list’.



20. In the reset password interface, you can change your password by entering your current password and new password.
The address is ‘http://localhost:3000/reset’.



6. References

1. Payne, A., & Frow, P. (2005). A Strategic Framework for Customer Relationship Management. Journal of Marketing, 69(4), 167-176.

2. Chen, I. J., & Popovich, K. (2003). Understanding customer relationship management (CRM): People, process and technology. Business Process Management Journal, 9(5), 672-688.

3. Marcus, A., & Gould, E. W. (2000). Cultural Dimensions and Global Web User-Interface Design: What? So What? Now What?. In Proceedings of the 6th Conference on Human Factors and the Web.

4. Ngai, E. W. T., Xiu, L., & Chau, D. C. K. (2009). Application of data mining techniques in customer relationship management: A literature review and classification. Expert Systems with Applications, 36(2), 2592-2602.