Software Engineering

Project title: Take away Self-Container in Colleges and Universities

Team members: Klaus, Anna, Bruce, Ivy, Olivia, Kathy.

URL:

https://github.com/Hexbruce/project-one
https://github.com/yumao-anna/project-1
https://github.com/OliviaWY29/software-engineering-
https://github.com/Ivy915/project-one
https://github.com/klaus668/software-engineering-project
https://github.com/kzh0726/SE/blob/master/usecase.docx

Catalogue:

Customer Problem Statement

a. problem statement

I am a student of Sias. I met some problems in campus life that brought me great inconvenience. In order to better manage the campus express, the school had set up some express delivery points, so I had to pick up my expressage at the express delivery points. You know the campus of Sias is very big. The worst thing is that the delivery points in our school are very scattered, and each express delivery point has its own running time. Last semester, I bought a lot of books on the Internet. The seller sent me express from different courier companies, such as ZTO Express, YTO Express, SF Express and so on. I had to rush around at different express delivery points on campus to get my expressages. Especially in the peak period of online shopping, such as the Double 11 shopping carnival, every express point had a long queue. I even stood in line for 20 minutes at one delivery point last year. Sometimes it takes so long time to pick up the expressage that I'm late for class. Because of the conflict between my class time and business hours of express delivery points, I cannot get my express delivery in time. Such things always make me tired.

Because of these reasons, I very much hope that there is a 24-hour the self-pick-up cabinet under dormitory buildings. The courier will deliver it to the self-pick-up cabinet under each dormitory building according to the dormitory building number on the shipping address. All my expressage will be put in this cabinet. When my expressages is placed in self-pick-up cabinet, I will receive a message containing the pick-up password (which can also be viewed in small programs or applications). I can

use the self-pick-up cabinet under the dormitory building at any time, and enter the pick-up password to take away my expressages. I don't have to worry about waiting for a long time to pick up the express, losing the expressages or missing the business hours of the express point.

On the other hand, I have a very tight schedule this semester, 11:40 A.M. after class, and 1:00 P.M. will be class. This an hour is the peak time for dining in the canteen. In order to have lunch as soon as possible I often choose to order a good takeaway and let the deliver guy put the takeout downstairs so that when I get back to my dormitory, I can have lunch at once. That will give me more time to do some other things, instead of wasting my time waiting in line in the canteen. But these takeaways do not have a reasonable place to lay up them. I found that my takeaway was taken away by someone else when I returned to the dormitory after class. I cannot find my lunch so that I have to go to the afternoon class hungrily. This is not uncommon, I often hear complaints about this situation, but so far there is no good solution.

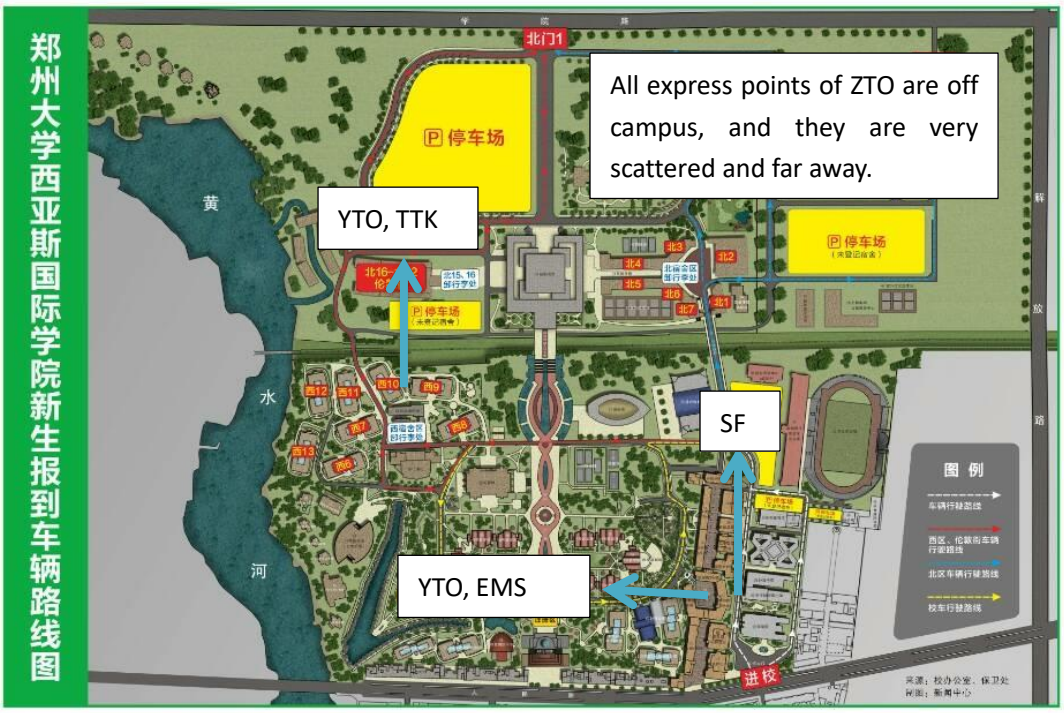
I hope the self-pick-up cabinet can also be used in takeaway storage, I can book a cupboard at the self-cabinet software in advance to store my takeaway and get a password for opening the self-pick-up cabinet, and then when I order the takeaway on line, just need to note my opening password to seller, so that when the deliver guy can put my takeaway into the self-pick-up cabinet. When I leave class at noon, I can go straight to the self-pick-up cabinet to get my takeaway and then end the rental of this self-pick-up cabinet. If this situation was true, I would no longer to worry about my takeaway being taken away by someone else.

In this software, I hope that its page design can be simple and clear, so that I can know at a glance what it is expressing or what should I do. And the software is divided into three parts. The first part is to receive express or takeaway. First, courier or takeout personnel use the cabinet or application program (mobile phone number + password) to authenticate, then scan the express list to confirm my mobile phone number, and select the size of the cabinet (takeout can choose takeout cabinet with thermal insulation function), and finally the cabinet door opens and closes automatically to complete storage. I can input the pick-up code or two-dimensional code on the cabinet display screen. The system will automatically open the cabinet door after successful verification in the background, and then I can take my takeaway or expressages.

The second part is to send express, I can choose to send express in the cabinet or application program, enter my ID number + destination address information of express delivery, select the size of cabinet, and then cabinet automatically opens the cabinet door to complete storage. When my express is stored, the courier will get my express message and password. Then he used the courier code to open the cabinet door to collect the courier and sent the courier number to me by the software.

The third part is self-use, I can freely choose how long l wants to use, and it required advance reservation before each use. I can book the self-pick-up cabinet on the mobile phone application, and then choose the size and utility time of the self-pick-up cabinet l want to use. When making an appointment, it needs to input the user's cell phone number so that the application can send the cabinet password to me.

During this period of time after the reservation successfully, I can through this password to use self-pick-up cabinet at any time. We should pay special attention to that after exceeding the appointed time; the fee will be added in a certain proportion, so it require to pick up on time.



b. Glossary of Terms

Glossary of Terms

Word	Explain
Quick landing	Users can enter their mobile phone number, authentication code and choose their identity to login directly without registering any more.
Account password login	Users login with registered mobile phone number and password
Forget the password	When a user logs in, the password set at the time of registration is forgotten.

Register	User's Behavior of Obtaining Accounts
My	User center
Cabinet situation	Is the cabinet empty?
Time	Use time of cabinet
Express	What others have sent you use the cabinet
Self use	Use the cabinet by yourself
Heat preservation	Keep the temperature of the items in the cabinet
Confirm	Confirm the current page information
Verification	Open the corresponding cabinet by two-dimensional code or verification code
News	Use Information and Activity Notification of Cabinets
Feedback	Problems in the Use of User Responses
Cancel	Cancel the selected option
Button	Verification button for accessing things on the cabinet
Position	Location of the cabinet being used by the user

2. System Requirements

a. Enumerated Functional Requirements

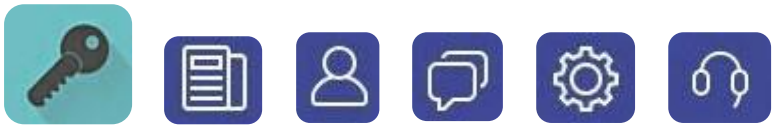
REQ' T	Priority weight	Customer requirement description	Function realization
REQ-1	**2**	**Users need to register accounts.**	**User registration**
REQ-2	**2**	**Users need to cancel their accounts.**	**Remove user**
REQ-3	**2**	**Users need account passwords to log in.**	**Login**
REQ-4	**2**	**Users need to log out.**	**Logout**
REQ-5	**5**	**Users need a way to store packages.**	**Sending two-dimensional code**
REQ-6	**3**	**Users need to choose whether to use the incubator or not.**	**Fill Order**
REQ-7	**3**	**Users need to fill in recipient information.**	**Fill Order**
REQ-8	**5**	**Users need to open the cabinet to store packages.**	**Send code**
REQ-9	**5**	**Users need a way to pick up packages.**	**Scan code**
REQ-10	**4**	**The system needs to charge the order fee.**	**Payment order**
REQ-11	**2**	**The system needs to receive user feedback for improvement.**	**Evaluation**
REQ-12	**8**	**The system requires data control center to record data.**	**Data processing**

b. Enumerated Nonfunctional Requirements：

We all know that the enumerated nonfunctional requirement contain FURPS. The F means function, the U means usability, the R means reliability, the P means performance and the S means supportability. First, I want to explain the Function, we just want to make an APP to help students and teachers to get the takeout, because some of the students and teachers would lose their takeout, and make their angry, causing the bad mood of having lunch. About the security, the system just sends the private messages to the users; other users cannot see the message. We ensure our system is security. We will build up the database to store our information about the students and teachers. Only users who need self-container can see only scream privately. Second, I want to explain the Usability. It is easy for students and teachers to use the APP, because we just receive the message and go to the self-container to take the takeout and close the door of the self-container. We will buy some beautiful self-container to let user feel comfortable. When we finish the transaction, our system will record the information. We can record the time, times, and the honest and break. Third, I want to explain the Reliability. When the system is broken, we can fix it in the computer. The courier who put the takeout to the self-container, the message can send the user's phone, it can ensure the accuracy. Forth, I want to explain the Performance, our system can distinguish whether the self-container is empty or is full, also, our system can distinguish whether the self-container's door is open or is close, and our system can distinguish whether the message send is successful or is fail. When the courier put the takeout in the self-container, the message will send the users' phone as soon as possible, also, the courier can use the APP to check which self-container is empty or is full as soon as possible. Fifth, I want to explain the Supportability. We can test the APP for one week, and we will make a report to send the users' phone to know the suggestion and bug. We would create the Android and IOS, so we can ensure the compatibility. About the configurability, the intelligence phone can suppose the APP. We can upload the erection sequence to the Internet, so all of the users can find the APP's erection sequence. About the scalability, we can add the advertisement to help the business, and we can charge extra fees.

c. User Interface Requirements

There should be five requirements for the interface. In order of priority, they are Key, Order List, Mine, Message, Setting and Customer service.



Key

Users must pass the validation before opening the cabinet, they need a QR code

or other verification code to open the cabinet. Users will get one from the “Key”.

Order List

The user interface should have the ability to enable users to query their orders. Users are supposed to know the status of their current and historical orders, they can know about them by the “Order List”.

Mine

There certainly is the option of entering the user center in the user interface. Users can browse, add, modify and query their account information by clicking “Mine”.

Messages

There may be transactions between users, so that communication between users is necessary. Users can send messages to others through this function.

Setting

Software settings are important for an APP. Users should have the right to manage software Jurisdictions in the setting center. Also, users can personalize their APP by setting it up.

Customer service

Users may give the operation team their opinions, and the team also needs feedback from customers. Customer service provides a way for customers to communicate with the operation team. Customer service is also responsible for resolving disputes arising from transactions.

3. Functional Requirements Specification

a. Stakeholders

Users：Students, teachers, courier and takeaway

Managers: “Campus Cabinet” Program Administrators

b. Actors and Goals

Actor	Actor‘s goal’	User case name
User	To create a new user account and Allow the use of self-contained cabinets	User registration(UC-1)
User	To retire an existing user account and disable access.	Remove user(UC-2)
User	To login software using account password	User login(UC-3)
User	To quit using the software	Logout(UC-4)
User	Processing all data received	Data processing(UC-5)
User	To fill in information about using cabinets	Fill Order(UC-6)
User	To Close the cabinet door	Close the cabinet door (UC-10)
User	To select the cabinet to use and obtain the storage code of the corresponding cabinet.	Send store code(UC-7)
User	Scanner Scanning storage Code on the Cabinet	Scan store code(UC-8)

User	To Get Pickup code for Use Cabinets	Get the Pickup code (UC-12)
User	Scanner scans the corresponding pick-up code	Scanning Pickup code (UC-13)
User	To pay orders generated by using cabinets	Payment order(UC-15)
User	To Evaluate the Feeling of Using Cabinets	User evaluation(UC-16)
Cabinet	To create a new user account and Allow the use of self-contained cabinets	User registration(UC-1)
Cabinet	To select the cabinet to use and obtain the storage code of the corresponding cabinet.	Send store code(UC-7)
Cabinet	The cabinet opens the corresponding cabinet door automatically	Open cabinet(UC-9)
Cabinet	Scanner Scanning storage Code on the Cabinet	Scan store code(UC-8)
Cabinet	Restoration of cabinet usability	Remove Occupancy Mark (UC-14)
Store code	To select the cabinet to use and obtain the storage code of the corresponding cabinet.	Send store code(UC-7)
Store code	Scanner Scanning storage Code on the Cabinet	Scan store code(UC-8)
Pickup code	To Get Pickup code for Use Cabinets	Get the Pickup code (UC-12)
Pickup code	Scanner scans the corresponding pick-up code	Scanning Pickup code (UC-13)
Scanner	Scanner Scanning storage Code on the Cabinet	Scan store code(UC-8)
Scanner	Scanner scans the corresponding pick-up code	Scanning Pickup code (UC-13)
Timer	To counts how often cabinets are used	Close the cabinet door (UC-10) Open cabinet(UC-9)

Memorizer	To store all information	UC-(1~16)

Initiating actor: User
Participating actor: Cabinets、Store code、Scanner、Timer、Pickup code、Order、Software

c. Use Cases

i: Casual Description:

UC-1. **User registration:** Users fill in personal information (name, phone, e-mail, ID number) to register an account.

UC-2. **Remove user:** The user logs out the account and no longer uses the system. Clear database personal information.

UC-3. **Login:** The user logs in to the program by using the registered account password.

UC-4. **Logout:** User quits personal account.

UC-5. **Data processing:** Responsible for the storage and processing of all system information.

UC-6. **Fill Order:** The sender fills in the order information (recipient's name, phone number, "box" ID), and choose whether you need an incubator.

UC-7. **Send store code:** The system receives the order information, and sends the QR code back.

UC-8. **Scan store code:** The cabinet scans the QR code.

UC-9. **Open cabinet:** The cabinet recognizes the QR code successfully and opens the cabinet.

UC-10.**Close cabinet:** The depositor puts the package and closes the door

UC-11.**Cabinet occupancy mark:** The data processing center marks that the cabinet has been used.

UC-12.**Send pick-up code:** The system receives the order information, and sends the pickup QR code to the id according to the order information.

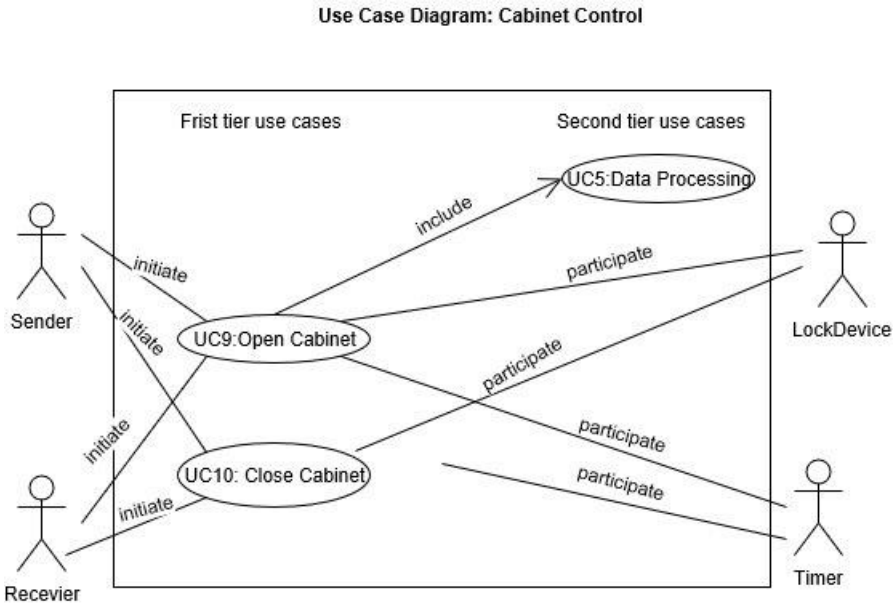
UC-13.**Scan pick-up code:** The cabinet scans the QR code.

UC-14.Cancel occupancy mark: After taking out the courier, the data control center cancels the occupancy mark.

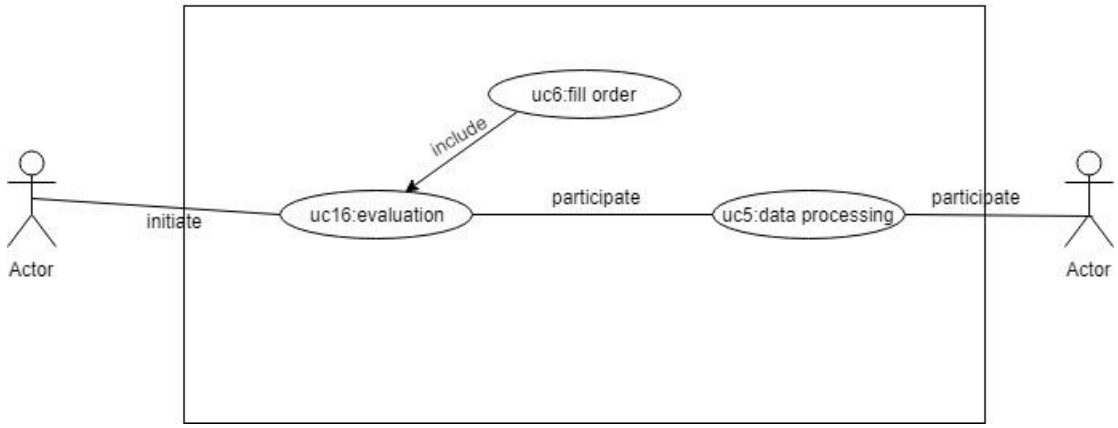
UC-15.**Payment order:** The system sends the order information to the receiver message box, and the receiver the order.

UC-16.**Evaluation：** The order is completed and the user can evaluate the order.

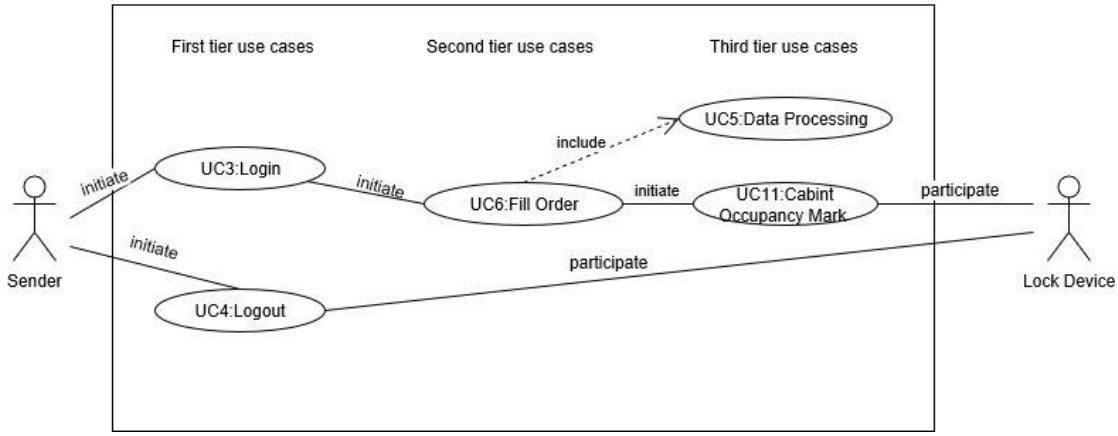
ii: Use Case Diagram:

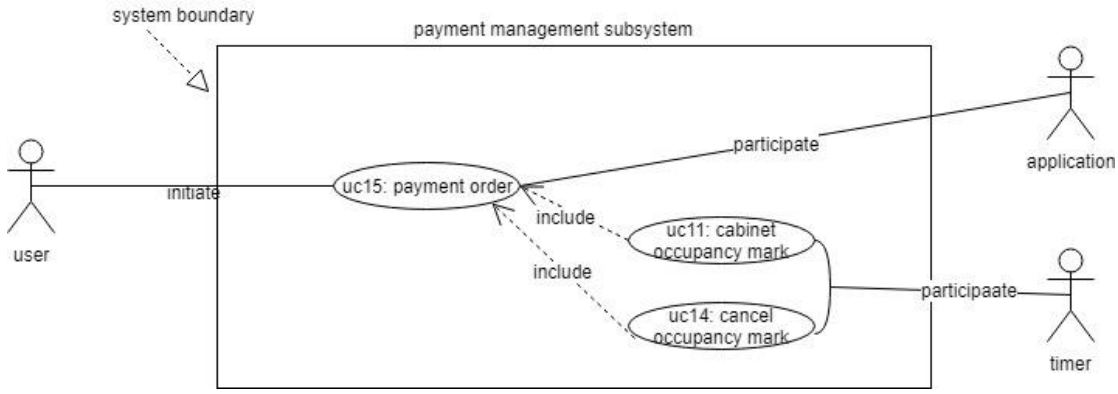


Use Case Diagram: evaluation control

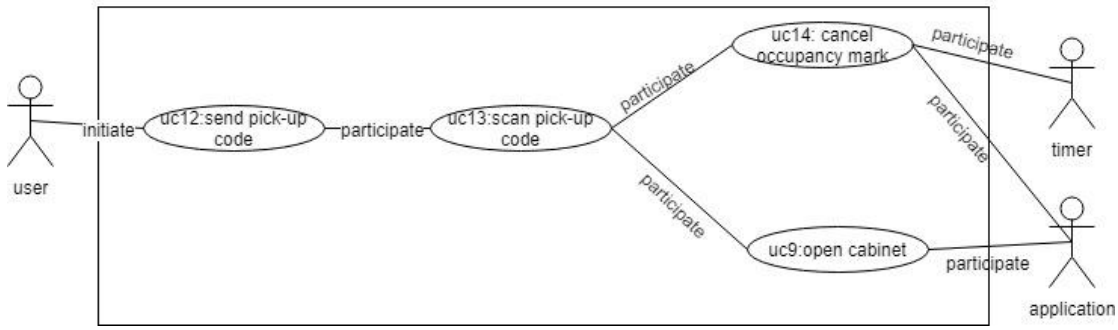


use case diagram:Order

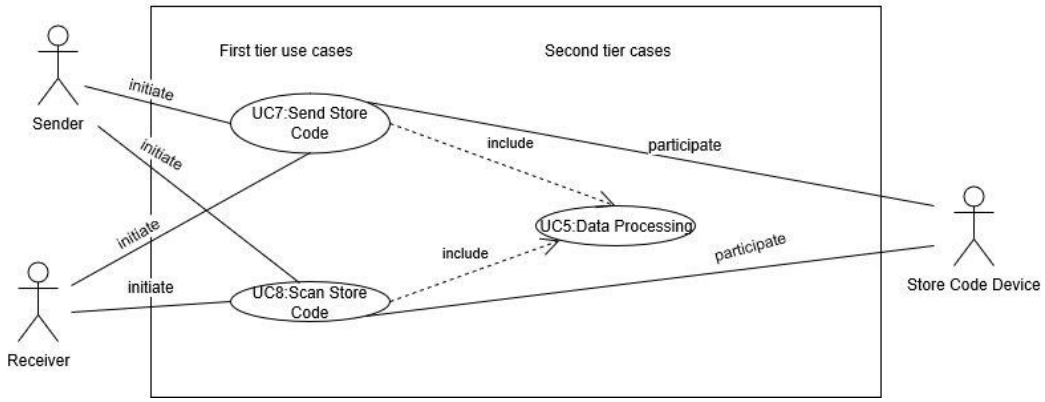


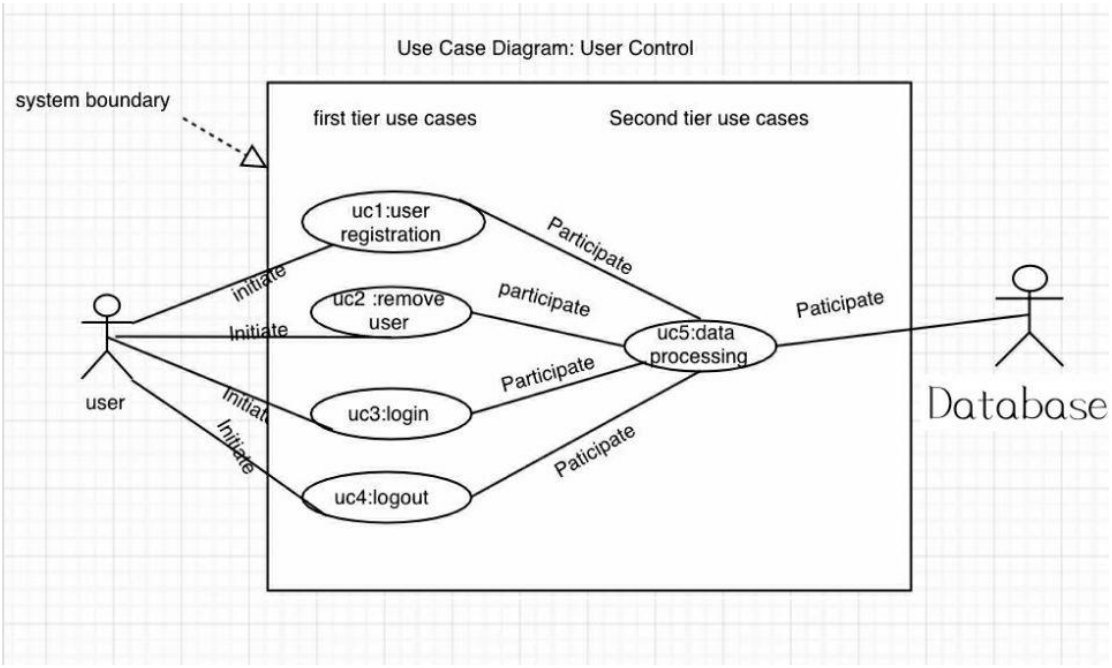


Use Case Diagram: pick-up control



Use Case Diagram: Store Code





iii: Traceability Matrix:

table1-2

Req't	PW	UC-1	UC-2	UC-3	UC-4	UC-5	UC-6	UC-7	UC-8	UC-9	UC-10	UC-11	UC-12	UC-13	UC-14	UC-15	UC-16
req-1	2	×				×											
req-2	2		×			×											
req-3	2			×		×											
req-4	2				×	×											
req-5	5					×	×	×	×	×							
req-6	3					×	×					×					
req-7	3					×	×					×					
req-8	5					×		×	×	×	×						
req-9	5					×					×		×	×	×		
req-10	4					×										×	
req-11	2					×											×
req-12	8					×											
Max PW	8	2	2	2	2	8	5	5	5	5	5	3	5	5	5	4	2
Max PW	43	2	2	2	2	43	11	10	10	10	10	6	5	5	5	4	2

UC-1: User registration: Users fill in personal information (name, phone, email, ID number) to register an account.
UC-2: Remove user: The user logs out the account and no longer uses the system. Clear database personal information.

UC-3: Login: The user logs in to the program by using the registered account password.
UC-4: Logout: User quits personal account.
UC-5: Data processing: Responsible for the storage and processing of all system information.
UC-6: Fill Order: The sender fills in the order information (recipient's name, phone number, "box" ID), and choose whether you need an incubator.
UC-7: Send store code: The system receives the order information, and sends the QR code back.
UC-8: Scan store code: The cabinet scans the QR code.
UC-9: Open cabinet: The cabinet recognizes the QR code successfully and opens the cabinet.
UC-10: Close cabinet: The depositor puts the package and closes the door
UC-11: Cabinet occupancy mark: The data processing center marks that the cabinet has been used.
UC-12: Send pick-up code: The system receives the order information, and sends the pickup QR code to the id according to the order information.
UC-13: Scan pick-up code: The cabinet scans the QR code.
UC-14: Cancel occupancy mark: Take out the courier, the data control center cancels the occupancy mark.
UC-15: Payment order: The system sends the order information to the receiver message box, and the receiver the order.
UC-16: Evaluation： The order is completed and the user can evaluate the order.

Enumerated Functional Requirements

Req-1 Users need to register accounts.
Req-2 Users need to cancel their accounts.
Req-3 Users need account passwords to log in.
Req-4 Users need to log out.
Req-5 Users need a way to store packages.
Req-6 Users need to choose whether to use the incubator or not.
Req-7 Users need to fill in recipient information.
Req-8 Users need to open the cabinet to store packages.
Req-9 Users need a way to pick up packages.
Req-10 The system needs to charge the order fee.
Req-11 The system needs to receive user feedback for improvement.
Req-12 The system requires data control center to record data.

iv: Fully-Dressed Description:

UC-1: User registration.

Use Case UC-1:User Registration
Related Requirements:REQ-1, REQ-3,REQ-12 Initiating Actor: User, Software Actor's Goal: To create a new user account and allow the use of self-contained cabinets Participating Actors: User, Memorizer, Preconditions: The system displays the menu of available functions. At the application the menu choice is Registration. Flow of Events for Main Success Scenario: ----->User opens the application and selects the menu item "Registration". include::User registration(UC-1), Data processing(UC-5) <-----a: system provide registration information order to fill out ----->b: the user fill in the personal information <-----c: system verify the account <-----d: system send the registration success or not to user e:data storage----->

UC-2: delete user.

Use Case UC-2： Remove User
Related Requirements:REQ-2 Initiating Actor: User, Software Actor's Goal: Delete user accounts and clear user data Participating Actors: User, Memorizer, Preconditions: The system displays the menu of available functions. At the application the menu choice is Remove Flow of Events for Main Success Scenario: ----->User opens the application and selects the menu item "Remove". include::Remove user(UC-2) <-----a: The system prompts the user to confirm whether to clear the user information ----->b: The user confirms to clear personal data from the database <-----c: system verify the account <-----d: The system sends a clear success message to the user e： Data to remove----->

UC-3: Login: Users log in with registered account passwords.

UC-3: Login: Users log in with registered account passwords.
Related Requirement's:REQ1-REQ12.table1-2 Initiating Actor: Any of: senders, recipients Actor's Goal: achieve the process of the log in, log out and registration. Participating Actors: data process control, system, information system Preconditions: none Post conditions: achieve the process of what the users want to get. Flow of Events for Main Success Scenario: ----→a. Users enter the login page and enter their account number and password for login. ←---b. The system receives the order and feedback to the users.

UC-4: Logout

Use Case UC-4:	Logout
Related Requirements:	REQ-4 stated in table 1-2
Initiating Actor:	Any of: senders and receivers
Actor's goal:	Achieve the process of exiting the software
Participating Actors:	System, Database
Preconditions:	The system is working and the user has logged in.
Post conditions:	None
Flow of Events for Main Success Scenario:	
----->a. User clicks the "MINE" button in the main interface.	
<-----b. System displays the personal center interface to the user.	
----->c. User clicks the "log out" button.	
<-----d. System prompted the user to ensure to log out.	
----->e. User clinks "Yes" to log out.	
<-----f. System stores the operate logs to database and shutdown.	

UC-5: data processing

UC-5: data processing
Related Requirement's:REQ1, REQ12, table1-2 Initiating Actor: Any of: senders, receivers Actor's Goal: Let the database system to store the database of the users. Participating Actors: name's students or teacher, address of the teacher and students,

name’s food and takeout, phone number, senders’ information Preconditions: The senders and receivers finished their order Post conditions: The data processing center store and arrange the database Flow of Events for Main Success Scenario: ----→a. The receivers finished the information, ←----b. The system receives this order, and request users to store the database ----→c. The users choose “yes” ←----d. The system stores the database and feedback to the receivers.

Use Case UC-（6）: Users fill in order information

Use Case UC-（6）: Users fill in order information
Related Requirement’s:REQ5, REQ5, REQ7,table1-2 Initiating Actor: Any of: senders Actor’s Goal: Ensuring accuracy and consistency of information Participating Actors: name’s students or teacher, address of the teacher and students, name’s food and takeout, phone number, order information, data processing center. Preconditions: Senders confirm the type of food or takeaway Post conditions: Senders confirm order information again, Flow of Events for Main Success Scenario: ---→ a. Senders determine the type of food and take-out, ←---b. the system displays order information to users, ---→c. users fill in order information. ←---d. The system feedback to the users.

Use Case UC-（7）Send storage code

Use Case UC-（7）Send storage code
Related Requirement’s:REQ5, REQ8, table1-2 Initiating Actor: All of: senders Actor’s Goal: let the sender to take the food or takeout into the cabinet. Participating Actors: data processing center, information system Preconditions: The receivers have finished the order information. Post conditions: The system get the behavior and send the storage code, and let the senders take the food of takeout into the cabinet. Flow of Events for Main Success Scenario: ---→a. The receivers wrote the order information, ←---b. the system understood the meaning and send storage code to the senders ---→c. the senders will use the storage code to open the cabinet’s door. ←---d. The door opened.

Use Case UC-（8）scan the storage code

Use Case UC-（8）scan the storage code
Related Requirement's: REQ5, REQ8, table1-2 Initiating Actor: All of: senders Actor's Goal: the senders want to take the food or takeout into the cabinet. Participating Actors: data processing center, canner, information system, Preconditions: The senders get the storage code Post conditions: The cabinet's door is opened. Flow of Events for Main Success Scenario: --→a. The senders get the code; he goes to the cabinet and scans the storage code. ←--b. The system confirms the behavior, and opens the door.

Use Case UC-（9）Open your own cabinet

Use Case UC-（9）Open your own cabinet
Related Requirement's:REQ5, REQ8, table1-2 Initiating Actor: Any of: recipients Actor's Goal: Get the own food and takeout Participating Actors: cabinet, door of the, scanner, system, data Processing Center Preconditions: Recipients got the fetch code and scan the QR code. Post conditions: The door opened. Flow of Events for Main Success Scenario: --→a. The recipients got the message and use the fetch code to scan the QR code. ←---b. The system will confirm the behavior. The recipients open the door.

Use Case UC-（10）Close own cabinet

Use Case UC-（10）Close own cabinet
Related Requirement's: REQ8, REQ9, table1-2 Initiating Actor: Any of: recipients Actor's Goal: close the own cabinet Participating Actors: cabinet, door of the cabinet, data processing center Preconditions: Recipients got the food or takeout successfully. Post conditions: The door is closed. Flow of Events for Main Success Scenario: --→a. The recipients got the food and takeout.

←--b. The system confirmed the behavior. The recipients need to close the door.

Use Case UC-（11） Tag occupation of self-contained cabinets

Use Case UC-（11） Tag occupation of self-contained cabinets
Related Requirement's: REQ6, REQ7, table1-2 Initiating Actor: Any of: senders, recipients, system Actor's Goal: the senders and recipients finished the behavior. Participating Actors: data processing center, cabinets Preconditions: The senders finished the behavior and close the door. Post conditions: The cabinets tag the occupation. Flow of Events for Main Success Scenario: --→a. The senders and recipients finished the process and close the door. ←--b. The system confirmed the behavior and tag occupation of the cabinets

Use Case UC-（12）System sending the fetch code

Use Case UC-（12）System sending the fetch code
Related Requirement's: REQ9, table1-2 Initiating Actor: System, fetching code Actor's Goal: Let users see their food or take-out information Participating Actors: recipients, system, data processing center Preconditions: System has received the successful order information, and sent the fetch code. Post conditions: Recipients received the fetch code. Flow of Events for Main Success Scenario: --→a. System saw the information and sent the fetch code ←--b. the recipients got the fetch code.

Use Case UC-（13）User scans the code

Use Case UC-（13）User scans the code
Related Requirement's: REQ9, table1-2 Initiating Actor: Any of: receivers Actor's Goal: Receivers get the food and takeout. Participating Actors: data processing center, system Preconditions: System got the access information about information code, and sent this code to the users

Post conditions: Users scan the door, and the door will open Flow of Events for Main Success Scenario: ---→a. The system got the information code and sent the access code to the recipients. ←---b. The recipients got the access code.

Use Case UC-（14） Cancel occupancy mark

Use Case UC-（14） Cancel occupancy mark
Related Requirements: REQ-6, REQ-9,REQ-12 Initiating Actor: Any of :User, Software, Actor's Goal: Restoration of cabinet usability Participating Actors: LockDevice, Timer, Memorizer, Preconditions: The LockDevice and code verification device of the cabinet are natural. User has been sent the pick-up code. Flow of Events for Main Success Scenario: include::Send pick-up code(UC-12), Scan pick-up code(UC-13), Cancel occupancy mark(UC-14) ----->a: User opens the application and selects the menu item"close the sale". <-----b: system provide the pick-up code to user. c:the system access database to verify the account information----->
<-----d: system cancel occupancy mark successful or not e:system store the operation logs to database----->

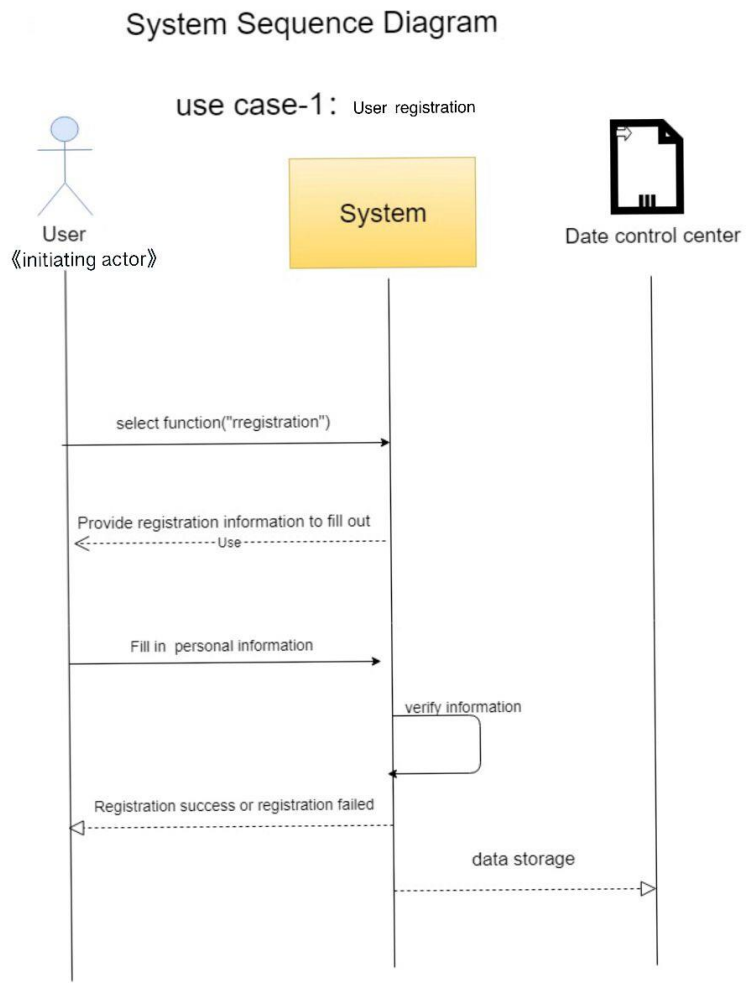
Use Case UC-（15）Payment order

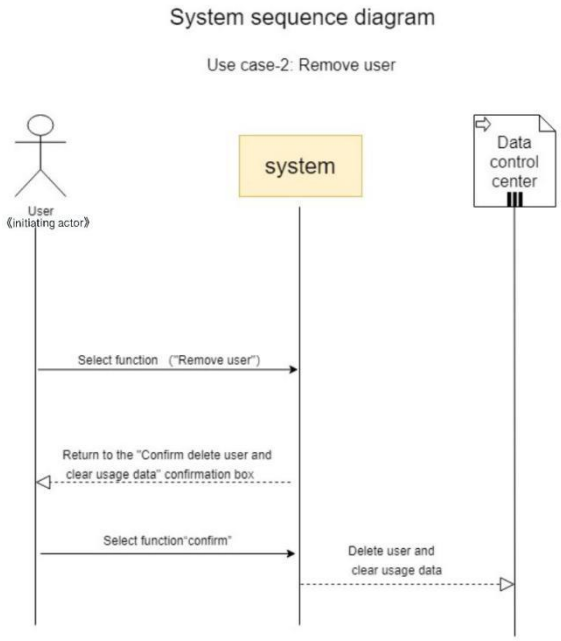
Use Case UC-（15）Payment order
Related Requirement's: REQ10, table1-2 Initiating Actor: All of: receivers Actor's Goal: The receivers finish the whole process. Participating Actors: data processing center, information system Preconditions: The receivers take the food and takeout Post conditions: The receivers finished the whole process. Flow of Events for Main Success Scenario: --→a. The receiver got the food and takeout. ←--b. The system confirmed the order information, and gave the payment order. --→c. The receivers pay money to this process. ←--d. The system record the data.

Use Case UC-（16）Evaluation

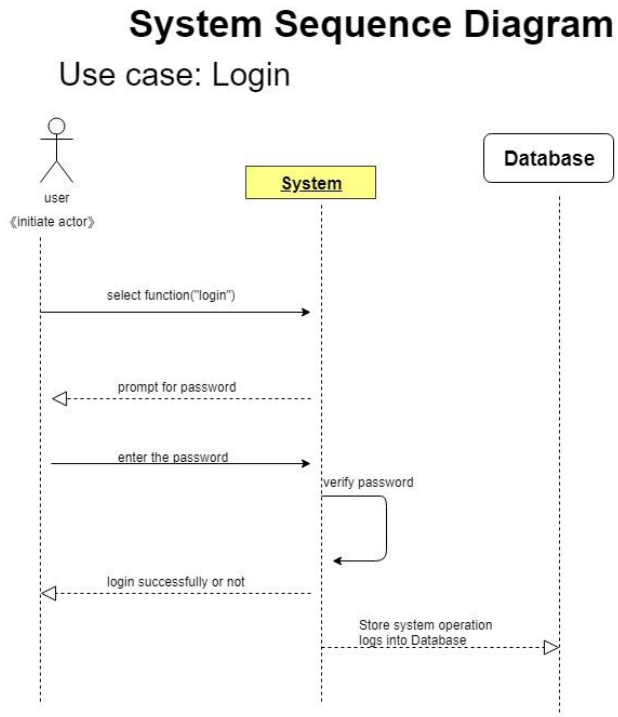
Use Case UC-（16）Evaluation
Related Requirement's: REQ11, table1-2 Initiating Actor: All of: receivers Actor's Goal: give the system feedback Participating Actors: Data processing center, evaluation order, information system Preconditions: The receivers wrote the evaluation order. Post conditions: none Flow of Events for Main Success Scenario: -→a. The receivers wrote the evaluation order ←--b. the system got the command, and got the feedback

d. System Sequence Diagrams:

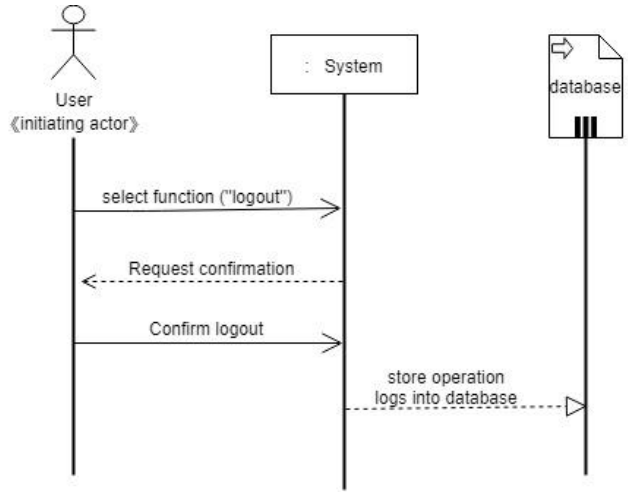



System sequence diagram
Use case-2: Remove user
User
《initiating actor》
system
Data
control
center
Select function ("Remove user")
Return to the "Confirm delete user and
clear usage data" confirmation box
Select function"confirm"
Delete user and
clear usage data

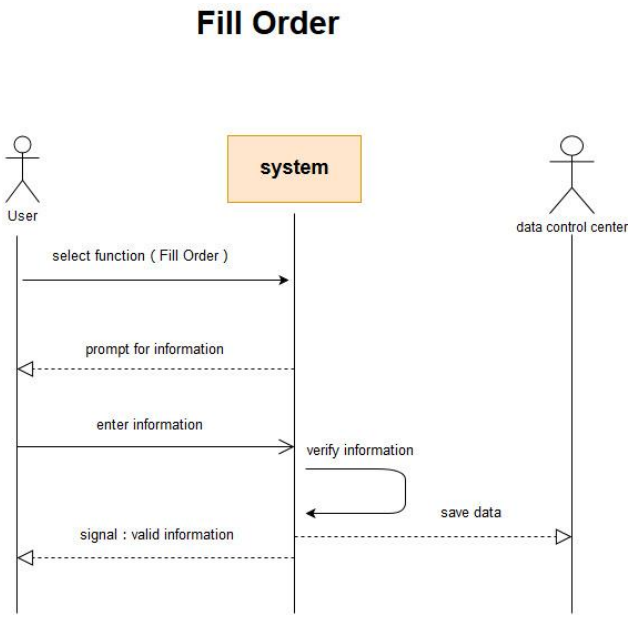
uc-3.


System Sequence Diagram
Use case: Login
user
《initiate actor》
System
Database
select function("login")
prompt for password
enter the password
verify password
login successfully or not
Store system operation
logs into Database

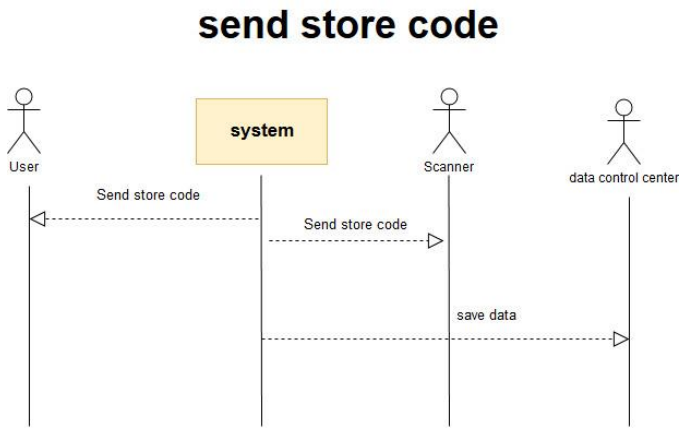
UC-4



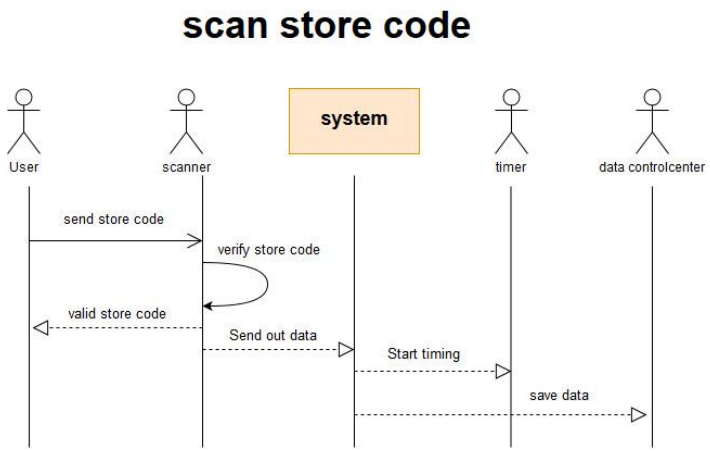
uc-6



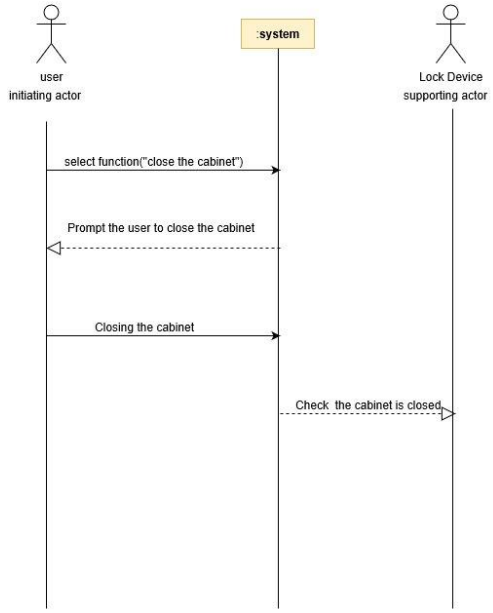
uc-7



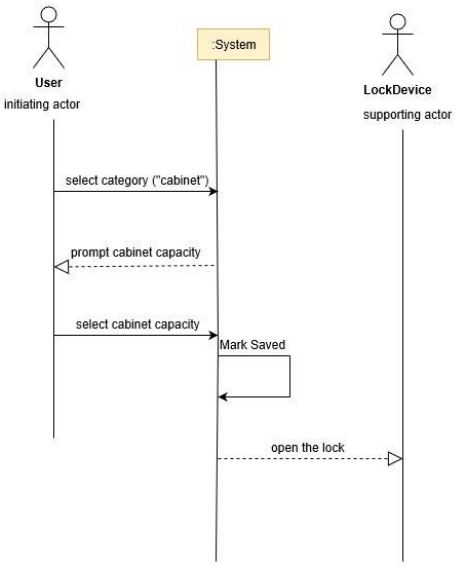
uc-8



Use Case 10 :close cabinet

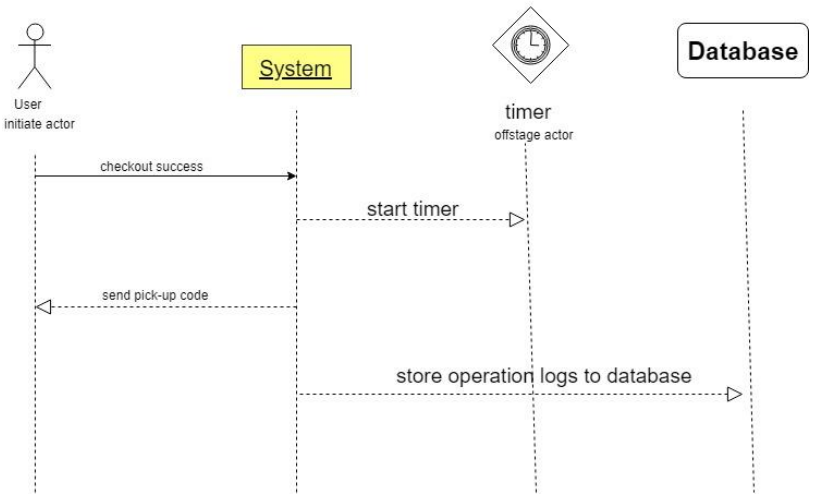


Use Case11:Cabinet occupancy mark

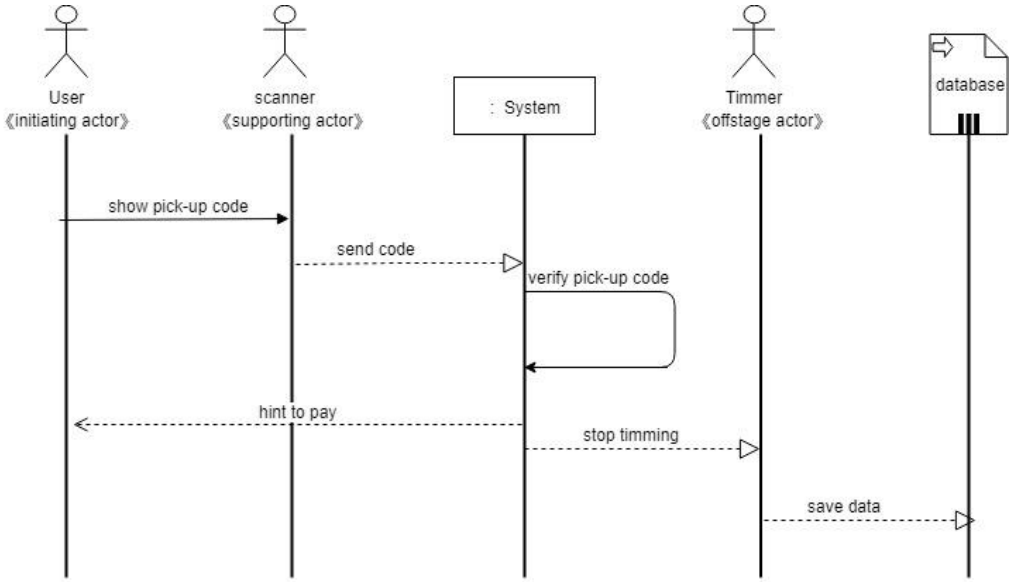


System Sequence Diagram

Use Case12: send pick-up code

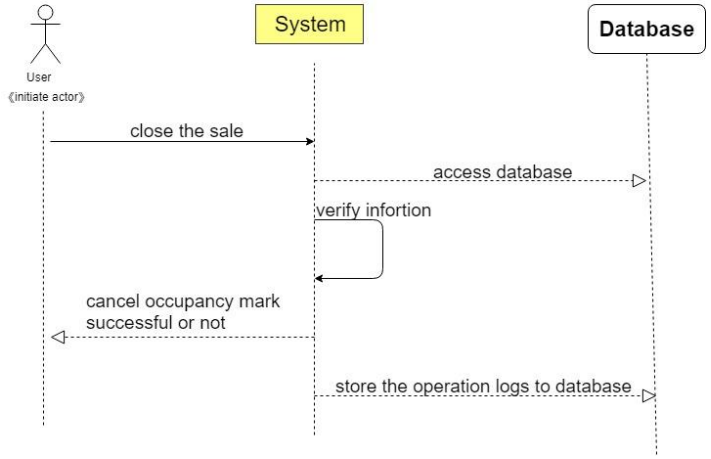


UC-13



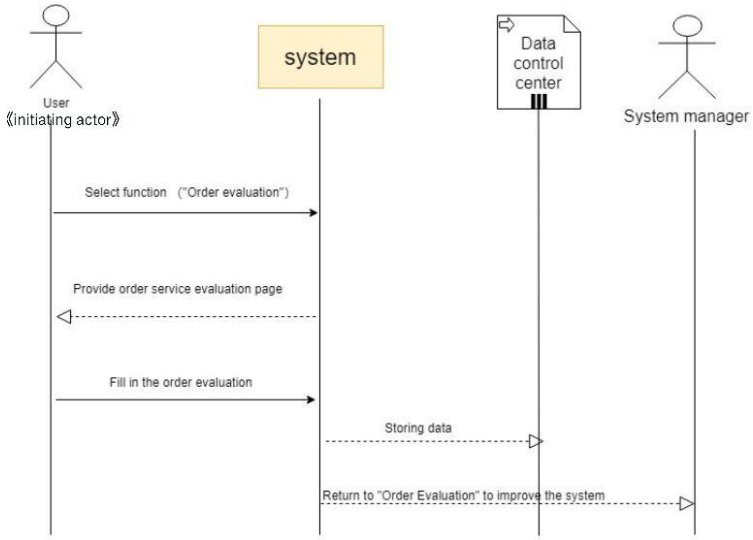
System Sequence Diagram

Use Case14:Cancel occupancy mark

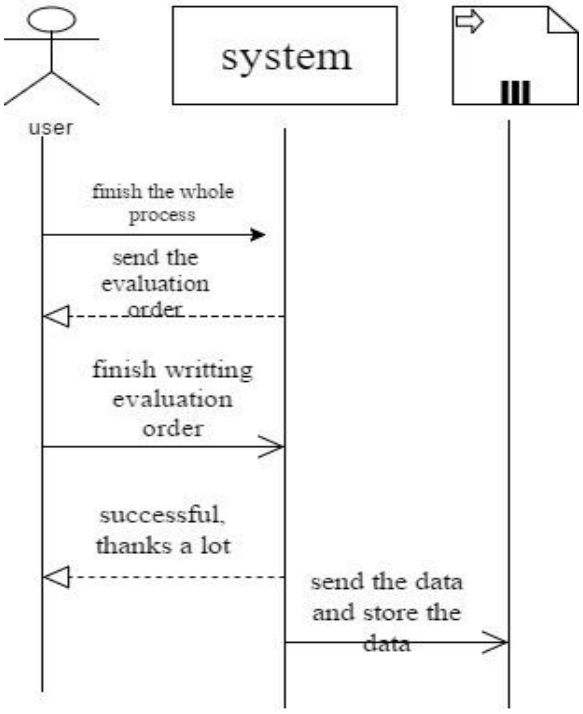


System sequence diagram

Use case-15: Payment order



UC-16 Evaulation



4. User Interface Specification

a. Preliminary Design:

UC-1 User registration

Users open the APP, the following login interface will be displayed. When the user have an account already, he can sign in, if not, he need to press the “Register” button to get one.

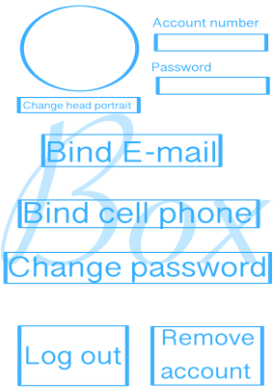
After the user presses the “Register” button, he will enter the registration interface to enter information. The user must enter his information in the appropriate box. Then click “confirm”.

If the user's information is correct, the user can register the account successfully. The user will get his account number. Now, the user can log

in by pressing “Sign in”.

UC2-Remove account

Users click the “MINE” button in the main interface to enter the personal center.



Users click the “remove account” button in the personal center interface to enter the remove confirmation interface.

After the user clicks “yes” on the confirmation interface, the account is removed.

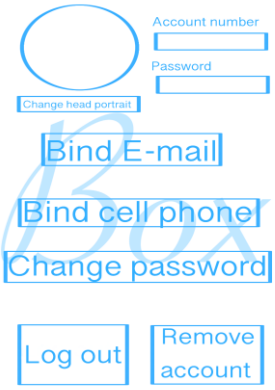
UC-3 Login

Users open the APP, the following login interface will be displayed. Users click the “sign in” to enter the log in interface.

After entering the correct user account and password, the user can press “sign in” to sign in.

UC-4 Logout

Users click the “MINE” button in the main interface to enter the personal center.



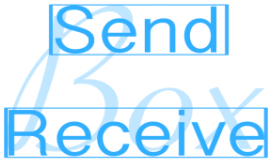
Users click the “log out” button in the personal center interface to enter

the log out confirmation interface.

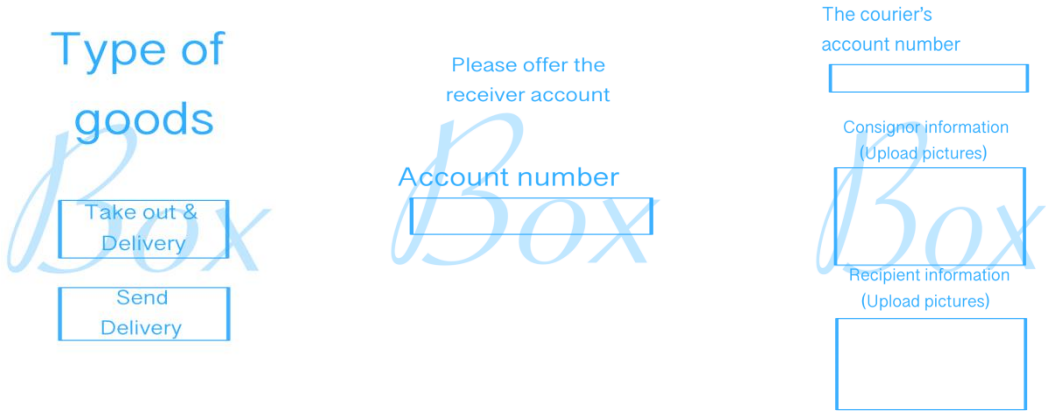
After the user clicks “yes” on the confirmation interface, he will log out.

UC6-Fill Order & UC-7 Send store code

The user clicks the “key” in the main interface to choose to send or receive.



The user click “send” to initiate a new order. Then choose the type of goods. Different types of goods to different Information Input interfaces.



After the user submits the receivers' information, he will get the store code to open the cabinet.

UC-12 Send pick-up code & UC-15 Payment order

The user clicks the "key" in the main interface to choose to send or receive.

The users click “receive” to get his pick-up code if someone else “send” any goods to him.

Users successfully take the goods, the interface automatically jumps to the payment interface.

UC-16 Evaluation

If the user wants to evaluate us or make suggestions, they can click the “service” in the main interface to enter our Customer service center.

b. User Effort Estimation:

UC-1: Registration:

A total of 54 clicks on the screen, as follows
Click the "Registration" button
Click the "Name" input box
Click “l”、”i”、”h”、”a”、”o”
Click the “Chinese ID” input box
Click ” x”、”x”、 ” x”、”x”、 ” x”、”x”、 ” x”、”x”、 ” x”、”x”、 ” x”、”x”、 ” x”、”x”、 ” x”、”x”
Click the “E-mail” input box
Click ” x”、”x”、 ” x”、”x”、 ” x”、”x”、 ” x”、”x”、 ” x”、”x”、 ” x”、”x”、 ” x”、”x”、 ” x”
Click the “Password” input box
Click ” x”、”x”、 ” x”、”x”、 ” x”、”x”、 ” x”、”x”
Click the "Confirm" button

UC-3: Sign in:

A total of 19 clicks on the screen, as follows
Click the "Sign in" button
Click the "User Account" input box
Click ” x”、”x”、 ” x”、”x”、 ” x”、”x”、 ” x”
Click the “Password” input box
Click ” x”、”x”、 ” x”、”x”、 ” x”、”x”、 ” x”、”x”
Click the "Sign in" input box

UC-9: Open cabinet: A total of 2 clicks on the screen, as follows

Click the "Key" button
Click the "Pick up" button

UC-4: Logout: A total of 3 clicks on the screen, as follows

Click the "My" button
Click the "Logout" button
Click the "Confirm" button

Participant sheet:

Bruce: Use Case (15): Casual Description, Diagram, Part of Traceability matrix, fully-Dressed Description.

Use Case (16): Casual Description, Diagram, Part of Traceability matrix, fully-Dressed Description

System Sequence Diagrams (4) (13) (16)

URL: https://github.com/Hexbruce/project-one

Anna: Use Case (4): Casual Description, Diagram, Part of Traceability matrix, fully-Dressed Description

Use Case (14): Casual Description, Diagram, Part of Traceability matrix, fully-Dressed Description.

System Sequence Diagrams (3) (12) (14)

URL: https://github.com/yumao-anna/project-1

Olivia: Use Case (7): Casual Description, Diagram, Part of Traceability matrix, fully-Dressed Description

Use Case (8): Casual Description, Diagram, Part of Traceability matrix, fully-Dressed Description

Use Case (9): Casual Description, Diagram, Part of Traceability matrix, fully-Dressed Description.

System Sequence Diagrams (8) (7)

ORL: https://github.com/OliviaWY29/software-engineering-

Ivy: Use Case (1): Casual Description, Diagram, Part of Traceability matrix, fully-Dressed Description

Use Case (2): Casual Description, Diagram, Part of Traceability matrix, fully-Dressed Description

Use Case (3): Casual Description, Diagram, Part of Traceability matrix, fully-Dressed Description

System Sequence Diagrams (1) (6)

URL: https://github.com/Ivy915/project-one

Kathy: Use Case (10): Casual Description, Diagram, Part of Traceability matrix, fully-Dressed Description

Use Case (11): Casual Description, Diagram, Part of Traceability matrix, fully-Dressed Description

Use Case (12): Casual Description, Diagram, Part of Traceability matrix, fully-Dressed Description.

System Sequence Diagrams (15) (11)
URL: https://github.com/kzh0726/SE/blob/master/usecase.docx

Klaus: Use Case (5): Casual Description, Diagram, Part of Traceability matrix, fully-Dressed Description
Use Case (6): Casual Description, Diagram, Part of Traceability matrix, fully-Dressed Description
Use Case (13): Casual Description, Diagram, Part of Traceability matrix, fully-Dressed Description
System Sequence Diagrams (2) (10)
URL: https://github.com/klaus668/software-engineering-project