

CS3343 Software Design

Bug Report

Group 12

|  |  |  |
| --- | --- | --- |
| Name | Student ID | Title |
| CHENG Yin | 56614557 | Project Manager |
| SONG Tao | 56642520 | Assistant Project Manager |
| LUO Peiyuan | 56642728 | Programmer |
| ZHOU Junchen | 56641511 | Programmer |
| DU Wenxi | 56643530 | Tester |
| FENG Yong | 56642741 | Tester |

**Table of contents**

[**1 Introduction**](#_1dyyq7288i38) **3**

[**2 Bug Repository**](#_yzcsja6ve03c) **3**

[**3 Guidelines for Bug Reporting**](#_jmqjqzr1klhb) **4**

[**4 Bug Reports**](#_bh2g1oaf97r6) **6**

[**4.1 General**](#_lronqp5x5ejm) **6**

[Issue #G1](#_hhldkvdcvlx4) 6

[Issue #G2 ​​](#_1f2t7h6xkme8) 7

[Issue #G3](#_yvq3zr20c5p7) 8

[Issue #G4](#_8rcu4gagts4y) 9

[Issue #G5](#_kmpeyuw63nhr) 10

[Issue #G6](#_h4p8b5vzs7yr) 11

[Issue #G7](#_lq3k3k6k5mcs) 12

[Issue #G8](#_ktynpuofwhaf) 13

[Issue #G9](#_s1bajy17r7i1) 14

[Issue #G10](#_nf8nsad07ao2) 15

[**4.2 Customer-related issues**](#_m7jz2okicufl) **16**

[Issue #C1](#_9wpopttejfg3) 16

[Issue #C2](#_37ez3n9f3a6i) 17

[Issue #C3](#_i4eeiz43rty) 18

[Issue #C4](#_yb7ryvbjwa45) 19

[Issue #C5](#_l3xrffcm26e) 20

[Issue #C6](#_4s1b0qkly7hu) 21

[**4.3 Merchant-related issues**](#_fv9s61whvutg) **22**

[Issue #M1](#_v0i5n335pmvg) 22

[Issue #M2](#_2a0hqr55hm) 23

[Issue #M3](#_sao4vlnqptit) 24

[**4.4 Admin-related Issues**](#_510aekrephzc) **25**

[Issue #A1](#_8gezqchnov9o) 25

[Issue #A2](#_8y8hdu6jrs95) 26

[Issue #A3](#_c79w9jlnlysk) 27

# 

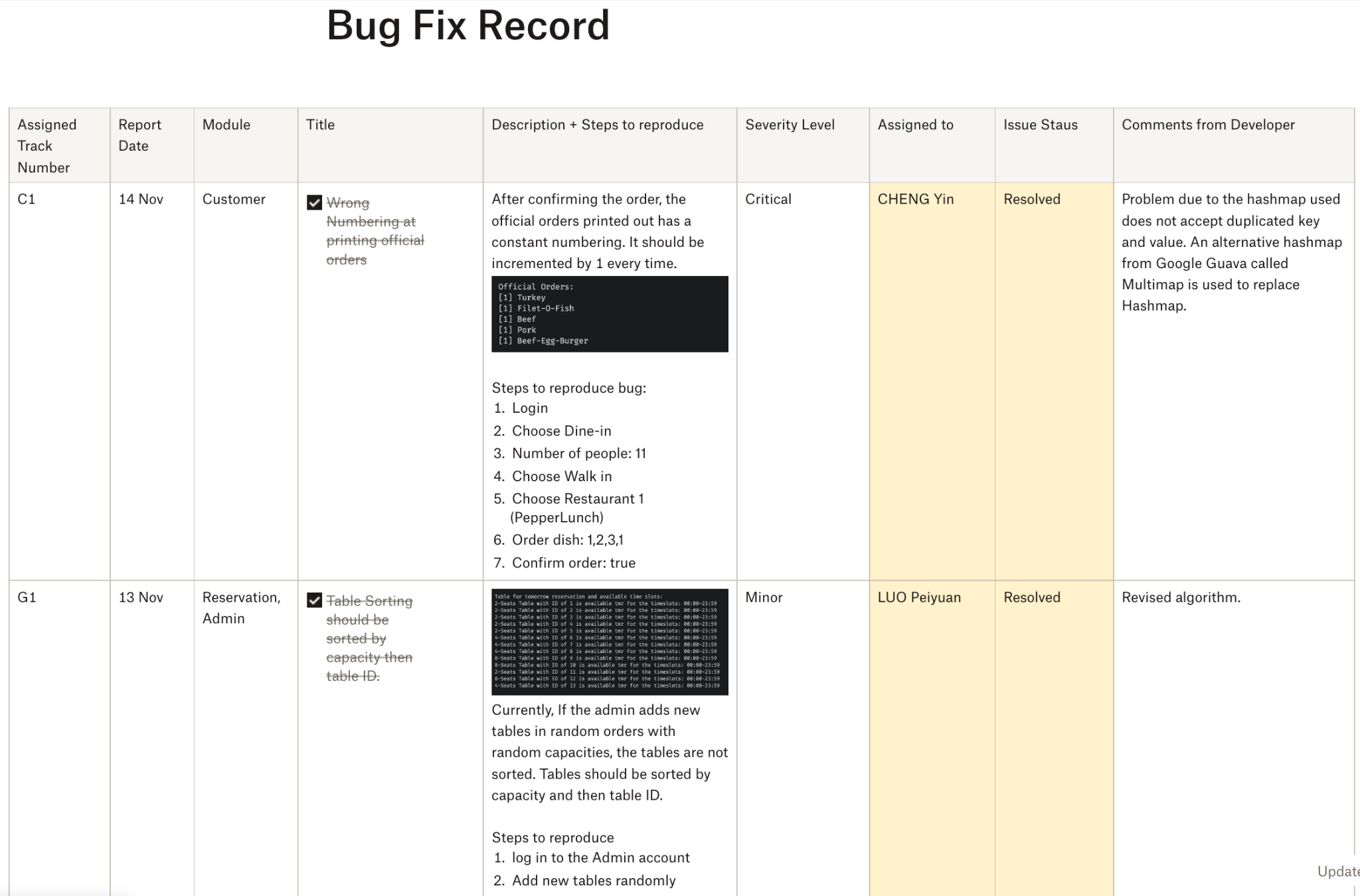
# **1 Introduction**

When the program is abnormal while testing, the testers will log the information in the Dropbox Paper, and after all, all the bug fixes and issues raised will be included in this Bug Report.

# **2 Bug Repository**

Dropbox Paper is the platform we used for logging bugs or abnormalities found.

In the Dropbox Paper, we created a paper named Bug Fix Record. Inside the paper, we created a table with categories highlighted. All the important items are marked inside the Bug Fix Record, and after the bug fixing period, the finished items will be summarized in the Bug Report.



# 

# **3 Guidelines for Bug Reporting**

There are important fields that testers must include in the Bug Fix Record; they are categories, titles, descriptions, steps to reproduce, and severity levels.

After the programmers receive the bugs raised in the Bug Fix Record, they will check if it is their responsibility for the classes. If they are responsible for that operation, they will write their names in the “Assigned” column and start working on it.

After checking and working on the problems raised by the testers, the developers will jot down the issue's status. And when finished, the developers should comment on what is leading to the bug and how they fix it to aid the bug fix if the problem happens again in the future.

Important Components of a bug report to be inputted by the tester:

* Title
* Report Date
* Module
* Description
* Steps to reproduce
* Severity Level

Categories of Issues:

|  |  |
| --- | --- |
| Category | Description |
| General | Bugs or abnormalities that do not belong to any of the admin-related, merchant-related, or customer-related issues. |
| Admin-related | Bugs or abnormalities that belong to the admin module. |
| Merchant-related | Bugs or abnormalities that belong to the merchant module. |
| Customer-related | Bugs or abnormalities that belong to the customer module. |

Severity Levels:

|  |  |
| --- | --- |
| Priority Level | Description |
| Minor | Typos etc., that do not affect the system |
| Moderate | Wrong output, will not interrupt the system running |
| Major | Works, but it is not the favorable behavior of the system |
| Critical | Unable to work and will interrupt system running |

Important Components of a bug report to be inputted by the developer:

* Assigned Developer
* Comment from Developer
* Issue Status

|  |  |
| --- | --- |
| Issue Status | Description |
| Open | After the bug has been raised from testers. |
| In Progress | After the bug has been assigned to a developer. |
| Resolved | After the developer has finished fixing the bug. |
| Closed | After checking the fixed bug by the tester. |
| Reopened | If the same problem remains from the resolved issue. |

# 

# **4 Bug Reports**

## **4.1 General**

### **Issue #G1**

|  |  |
| --- | --- |
| **Title:** | Table Sorting should be sorted by capacity, then table ID. |
| **Reported Date:** | 1 November 2022 |
| **Severity Level** | **Moderate** |
| **Module:** | Admin, Reservation |
| **Assigned To:** | **LUO Peiyuan** |
| **Status:** | Resolved |
| **Problem Description:**    Currently, If the admin adds new tables in random orders with random capacities, the tables are not sorted. Tables should be sorted by capacity and then table ID. | |
| **Steps to reproduce the bug:**   1. log in to the Admin account 2. Add new tables randomly 3. Logout 4. Log in to the Customer account 5. Choose reservation | |
| **Comment By Developer:**  Revised with collections sorting, changed from Collections.sort(copyOfAvailableTables); **to** Collections.sort(all, Collections.reverseOrder()); | |

### 

### **Issue #G2 ​​**

|  |  |
| --- | --- |
| **Title:** | Table ID does not exist but is still being printed |
| **Reported Date:** | 7 November 2022 |
| **Severity Level:** | **Minor** |
| **Module:** | Customer, Table |
| **Assigned To:** | **SONG Tao** |
| **Status:** | Resolved |
| **Problem Description:**    Table ID 343 does not exist, but I get this printed: “Can’t delete the table with id of 343. Maybe due to its unavailability, or it doesn’t exist.”  However, it should not output table id 343 since the table id does not exist. | |
| **Steps to reproduce the bug:**   1. Log in to the Customer account 2. Choose reservation 3. Input random time slot 4. Input table ID that does not exist | |
| **Comment By Developer:**  Revised the wording of the output. | |

### **Issue #G3**

|  |  |
| --- | --- |
| **Title:** | Need to add confirmation check in registering account |
| **Reported Date:** | 7 November 2022 |
| **Severity Level:** | **Major** |
| **Module:** | Registration |
| **Assigned To:** | **CHENG Yin** |
| **Status:** | Resolved |
| **Problem Description:**  Require a confirm function in register account  Else random input can create an account.  Current register process: | |
| **Comment By Developer:**  A ***confirmToRegister*** function in ***CommandAccountManagementRegister*** is added when registering a customer or merchant account. Only when the user confirms to register, the respective accounts will be registered. | |

### 

### **Issue #G4**

|  |  |
| --- | --- |
| **Title:** | Error! Official Order did not accept duplicate dish |
| **Reported Date:** | 8 November 2022 |
| **Severity Level:** | **Critical** |
| **Module:** | Customer, Dish |
| **Assigned To:** | **CHENG Yin** |
| **Status:** | Resolved |
| **Problem Description:**  Official Order should accept duplicate dish | |
| **Steps to reproduce the bug:**  Enter the system as a Customer, use dine in operation and take orders in a restaurant. Then order the same dish more than one time. | |
| **Comment By Developer:**  Hashmap used is replaced by Multimap to support duplicate keys. | |

### 

### **Issue #G5**

|  |  |
| --- | --- |
| **Title:** | Table allocation that does not meet the requirements |
| **Reported Date:** | 9 November 2022 |
| **Severity Level:** | **Critical** |
| **Module:** | Customer, Table |
| **Assigned To:** | **LUO Peiyuan** |
| **Status:** | Resolved |
| **Problem Description:**    When eight customers are inputted as the number of customers to dine in, the system gives out Two 4-seat tables and One 5-seat table for the table arrangement.  When eight customers come to dining, it is better to allocate two 4-seat tables. | |
| **Steps to reproduce the bug:** | |
| **Comment By Developer:**  In TableManagement, ***arrangeTableAccordingToNumOfPeople*** is revised to ***getTableArrangementResult.*** | |

### 

### **Issue #G6**

|  |  |
| --- | --- |
| **Title:** | The table id is not unique |
| **Reported Date:** | 10 November 2022 |
| **Severity Level:** | **Major** |
| **Module:** | Admin, Table |
| **Assigned To:** | **LUO Peiyuan** |
| **Status:** | Resolved |
| **Problem Description:**    Same table id being added twice. It should only be added once. | |
| **Steps to reproduce the bug:** | |
| **Comment By Developer:**  Added ***checkTableIdIsAreadyInUsed*** function in ***addNewTable*** to prevent the same table ID from being added twice. | |

### **Issue #G7**

|  |  |
| --- | --- |
| **Title:** | Reservation Time Slot length does not meet the requirements |
| **Reported Date:** | 11 November 2022 |
| **Severity Level:** | **Moderate** |
| **Module:** | Customer, Reservation |
| **Assigned To:** | **SONG Tao** |
| **Status:** | Resolved |
| **Problem Description:**  Reservation should have time restriction:  Cannot reserve time slot less than half an hour or more than 2 hours    New Update: need to print the error message: cannot reserve timeslot less than half hour or more than 2 hours | |
| **Steps to reproduce the bug:**  Enter the system as a Customer and go to the reservation page. Enter a time slot that the range is lesser than 30 mins or more than 2 hours | |
| **Comment By Developer:**  Added restriction to the time limit for reservation. | |

### **Issue #G8**

|  |  |
| --- | --- |
| **Title:** | **Table capacity cannot be removed when removing table** |
| **Reported Date:** | 16 November 2022 |
| **Severity Level:** | **Critical** |
| **Module:** | TablesManagement |
| **Assigned To:** | **LUO Peiyuan** |
| **Status:** | Resolved |
| **Problem Description:**    Integer ArrayList can not directly remove the value. | |
| **Steps to reproduce the bug:** | |
| **Comment By Developer:**  Fixed. | |

### 

### **Issue #G9**

|  |  |
| --- | --- |
| **Title:** | **Wrong return value for table reservation** |
| **Reported Date:** | 19 November 2022 |
| **Severity Level:** | **Moderate** |
| **Module:** | Table Management |
| **Assigned To:** | **LUO Peiyuan** |
| **Status:** | Resolved |
| **Problem Description:**  The function reserveTableAccordingToTimeslot() should return a boolean value when the timeslot has already been booked. | |
| **Steps to reproduce the bug:** | |
| **Comment By Developer:**  Fixed. | |

### 

### **Issue #G10**

|  |  |
| --- | --- |
| **Title:** | Exception handling in Admin setOpenCloseHour |
| **Reported Date:** | 20 November 2022 |
| **Severity Level:** | **Moderate** |
| **Module:** | Admin, Time management |
| **Assigned To:** | **SONG Tao** |
| **Status:** | Resolved |
| **Problem Description:**  In admin’s set opening hours operation, when input the wrong time format to change time, the system outputs the null error and should be handled with exception handling. | |
| **Steps to reproduce the bug:** | |
| **Comment By Developer:**  Exception handling is added. | |

## 

## 

## 

## **4.2 Customer-related issues**

### **Issue #C1**

|  |  |
| --- | --- |
| **Title:** | Wrong Numbering at printing official orders |
| **Reported Date:** | 1 November 2022 |
| **Severity Level:** | **Critical** |
| **Module:** | Customer |
| **Assigned To:** | **CHENG Yin** |
| **Status:** | Resolved |
| **Problem Description:**  After confirming the order, the official orders printed out has a constant numbering. It should be incremented by one every time. | |
| **Steps to reproduce the bug:**   1. Login 2. Choose Dine-in 3. Number of people: 11 4. Choose Walk in 5. Choose Restaurant 1 (PepperLunch) 6. Order dish: 1,2,3,1 7. Confirm order: true | |
| **Comment By Developer:**  Corrected the counting inside loop. | |

## 

### **Issue #C2**

|  |  |
| --- | --- |
| **Title:** | No option to exit reservation once executed |
| **Reported Date:** | 8 November 2022 |
| **Severity Level:** | **Moderate** |
| **Module:** | Customer, Reservation |
| **Assigned To:** | **CHENG Yin** |
| **Status:** | Closed |
| **Problem Description:**  There is no option to exit the reservation prompt when I executed it. It keeps prompting me to input the time string if my format is wrong. It should exit when I input “cancel”. | |
| **Steps to reproduce the bug:**   1. Login 2. Choose reservation | |
| **Comment By Developer:**  Not count into consideration | |

### 

### 

### **Issue #C3**

|  |  |
| --- | --- |
| **Title:** | **Wrong output for reservation** |
| **Reported Date:** | 11 November 2022 |
| **Severity Level:** | **Major** |
| **Module:** | **Customer, Reservation** |
| **Assigned To:** | **SONG Tao** |
| **Status:** | Resolved |
| **Problem Description:**  During every loop in running the customer’s module, if there exists reservation for customers, the output should be a reminder, not “Reservation made”. Only when the reservation is finished for the customer, it will output “Reservation made”.  Cannot reserve between 11:26-11:28, but still, output “reservation made” | |
| **Steps to reproduce the bug:**   1. Log in as customer 2. Select to reserve 3. Input 11:26-11:28 as time slot 4. Input table id 1 | |
| **Comment By Developer:**  Corrected output phrase set for finishing reservation. | |

### **Issue #C4**

|  |  |
| --- | --- |
| **Title:** | **Infinite Loop in exiting queue** |
| **Reported Date:** | 11 November 2022 |
| **Severity Level:** | **Critical** |
| **Module:** | Customer, Queue |
| **Assigned To:** | **CHENG Yin** |
| **Status:** | Resolved |
| **Problem Description:**  Infinite loop when trying to select to leave when needing to queue | |
| **Steps to reproduce the bug:**   1. Login as customer 2. Select to dine in 3. Input number greater than the max capacity of all tables, i.e., 32. 4. Choose to leave when prompted to queue or leave. | |
| **Comment By Developer:**  Corrected the output phrase set for ***recommendedArrangement***. | |

### 

### **Issue #C5**

|  |  |
| --- | --- |
| **Title:** | **Payment method cannot be shown** |
| **Reported Date:** | 15 November 2022 |
| **Severity Level:** | **Critical** |
| **Module:** | Customer, Payment |
| **Assigned To:** | **ZHOU Junchen** |
| **Status:** | Resolved |
| **Problem Description:**  Payment method name error  When payment is complete, the payment method is a string of gibberish. | |
| **Steps to reproduce the bug:** | |
| **Comment By Developer:**  Used toString() method. | |

### 

### **Issue #C6**

|  |  |
| --- | --- |
| **Title:** | **No need for a reminder for reservation if reservation not made** |
| **Reported Date:** | 15 November 2022 |
| **Severity Level:** | **Major** |
| **Module:** | Reservation |
| **Assigned To:** | **ZHOU Junchen** |
| **Status:** | Resolved |
| **Problem Description:**  If there is no reservation made, the program should not output the reservation reminder. | |
| **Steps to reproduce the bug:**  Customer Reservation: | |
| **Comment By Developer:**  Revised. | |

## 4.3 Merchant-related issues

### **Issue #M1**

|  |  |
| --- | --- |
| **Title:** | Merchant should not check customers’ orders that do not belong to his/her restaurant |
| **Reported Date:** | 2 November 2022 |
| **Severity Level:** | **Critical** |
| **Module:** | Merchant |
| **Assigned To:** | **CHENG Yin** |
| **Status:** | Resolved |
| **Problem Description:**  In the merchant module, when choosing to check customers’ orders, it printed a lot of dishes from orders that do not belong to the restaurant owned by the merchant. It should print the dish from orders that belong to the restaurant owned by the merchant only. | |
| **Steps to reproduce the bug:** | |
| **Comment By Developer:**  A multimap is added to store the dish corresponding to the restaurant. A function ***customerOrdersAccordingToRestaurant*** is used to return the corresponding dish of the merchant. | |

### **Issue #M2**

|  |  |
| --- | --- |
| **Title:** | **Wrong output for selection while choosing to modify menu.** |
| **Reported Date:** | 4 November 2022 |
| **Severity Level:** | **Major** |
| **Module:** | **Merchant** |
| **Assigned To:** | **CHENG Yin** |
| **Status:** | Resolved |
| **Problem Description:**  When going into the merchant’s modify menu, should output choice to add dish, remove dish, or edit dish, instead of outputting modifying menu again. | |
| **Steps to reproduce the bug:**   1. Login as Merchant 2. Choose Modify Menu | |
| **Comment By Developer:**  Corrected the output phrase set for modifying the menu. | |

### **Issue #M3**

|  |  |
| --- | --- |
| **Title:** | **Wrong Output for checking orders** |
| **Reported Date:** | 7 November 2022 |
| **Severity Level:** | **Minor** |
| **Module:** | Merchant |
| **Assigned To:** | **CHENG Yin** |
| **Status:** | Resolved |
| **Problem Description:**  Should be “This customer has no orders” not “No customer instance found” | |
| **Steps to reproduce the bug:**   1. Log in to the merchant’s account 2. Choose the operation [2] to check customers’ orders 3. Input the customer ID of whom do not have orders made 4. “No customer instance found” wordings are printed even when the customer id is printed out | |
| **Comment By Developer:**  Wrong exception handling, revised to print out no orders found. | |

## **4.4 Admin-related Issues**

### **Issue #A1**

|  |  |
| --- | --- |
| **Title:** | Account List printed is not sorted |
| **Reported Date:** | 28 October 2022 |
| **Severity Level:** | **Minor** |
| **Module:** | Admin |
| **Assigned To:** | **CHENG Yin** |
| **Status:** | Resolved |
| **Problem Description:**  As shown in the below screenshot, the active account list retrieved from admin module is not sorted by user ID. And the list’s formatting is a bit chaotic. Please align to the left with the numbering, user ID, and username. | |
| **Steps to reproduce the bug:**  At the start of the program. | |
| **Comment By Developer:**  A sorting algorithm in ***AccountManagement*** is added.  The sorting algorithm will first sort the prefix of the user ID in numerical order; then it will sort the id based on the last four digits. | |

## 

### 

### **Issue #A2**

|  |  |
| --- | --- |
| **Title:** | Wrong output in admin module when checking customers’ orders |
| **Reported Date:** | 6 November 2022 |
| **Severity Level:** | **Minor** |
| **Module:** | Admin |
| **Assigned To:** | **CHENG Yin** |
| **Status:** | Resolved |
| **Problem Description:**  When running admin’s operation to check customers’ orders, there is the “Official orders” wording printed out even if there are no customers’ orders returned.  It should not print “official orders” when there are no customers’ orders. | |
| **Steps to reproduce the bug:**   1. Log in to the admin’s account 2. Choose the operation to check customers’ orders 3. Input the customer ID of whom do not have orders made 4. Only “Official Orders” wordings are printed | |
| **Comment By Developer:**  Revised the output when there are no orders made. The revised version will print out “No orders made”. | |

### **Issue #A3**

|  |  |
| --- | --- |
| **Title:** | **Wrong output for Exception Handling in Checking order** |
| **Reported Date:** | 13 November 2022 |
| **Severity Level:** | **Minor** |
| **Module:** | Admin, Customer |
| **Assigned To:** | **CHENG Yin** |
| **Status:** | Resolved |
| **Problem Description:**    Should catch customer ID not found, not “d != java.lang.String” | |
| **Steps to reproduce the bug:**   1. Login as Admin 2. Choose to check customers’ orders 3. Input customer id that do not exist. | |
| **Comment By Developer:**  Added Exception handling to correct output if customer instance not found. | |

End of Report