

City University of Hong Kong

Department of Computer Science

CS3343 Software Engineering Practice

2024 - 25 Semester A

**CityU Cinema System**

**Bug Report**

**Group 2**

**Name SID**

CHAN Ka Hong 57845658

FUNG Wing Ho 56755919

LAW Lok Hang 57842515

NG Ka Ho 56652261

WONG Tsz Kin 57847086

**Report 1**

|  |  |
| --- | --- |
| **Title** | Get incorrect customer age during unit testing |
| **Date** | 2 November 2024 |
| **System Information** | Operating System: Windows 11 Home 23H2  OS build: 22631.4317  JDK version: 23 |
| **Severity** | Low |
| **Reported By** | Thomas |
| **Status** | Resolved |
| **Problem** | When I wrote the unit test cases for PaymentRecord, I used customer.getAge() to get the customer's age and assert whether the paymentRecord object was created successfully. However, the getAge() method returned 0 instead of the age I created. |
| **Steps to reproduce** | 1. Create a customer object 2. Use getAge() method |
| **Expected output** | The age passed in when creating a customer object  e.g. If I use the following code to create a customer object, it should return 10  Customer customer = new Customer("abc"password", 10);  Customer.getAge() // should return 10 |
| **Actual output** | It returns 0 instead of 10 |
| **Comments** | A buggy code that fails to initialize the customer age with the corresponding argument passed to the constructor. It was easily fixed by accepting and initializing the age attribute with the specified constructor’s argument. |

**Report 2**

|  |  |
| --- | --- |
| **Title** | Incorrect start time and end time of movie session |
| **Date** | 3 November 2024 |
| **System Information** | Operating System: Microsoft Windows 11 Home  OS build: 10.0.22631 Build 22631  JDK version: 21 |
| **Severity** | Severe |
| **Reported By** | Jasper |
| **Status** | Resolved |
| **Problem** | Following the integration of the movie scheduling module with the database data, when attempting the first execution of the program, the output of the scheduled movies was not as expected as both the start time and end time were abnormally shown. |
| **Steps to reproduce** | 1. Load default data to database  2. Create a scheduleManagement object  3. Schedule the movies based on the default schedule algorithm  4. Print out the scheduled movie infomation |
| **Expected output** | Both the start time and end time should be within the time limit of 00:00 to 24:00. |
| **Actual output** | Start time and end time range from 660:00 to 1439:50 |
| **Comments** | Misused some helper functions in time format conversion from string to int. Used the correct helper functions and then problem was solved. |

**Report 3**

|  |  |
| --- | --- |
| **Title** | NullPointerException: House Class |
| **Date** | 3 November 2024 |
| **System Information** | Operating System: Microsoft Windows 11 Home  OS build: 10.0.22631 Build 22631  JDK version: 21 |
| **Severity** | Low |
| **Reported By** | Jasper |
| **Status** | Resolved |
| **Problem** | “Exception in thread "main" java.lang.NullPointerException: Cannot invoke "java.util.List.iterator()" because the return value of "selectMovieModule.House.getMovieSessions()" is null  at MovieScheduling.Main.main(Main.java:36)”  Received the above exception message when trying to list all scheduled movie sessions by house. Suspected to not have initialized List<MovieSession> in the House class |
| **Steps to reproduce** | 1. Load default data to database  2. Create a scheduleManagement object  3. Schedule the movies based on the default schedule algorithm  4. Print out the scheduled movie information by House object |
| **Expected output** | Should list out all scheduled movie session by each house. |
| **Actual output** | A NullPointerException was issued. |
| **Comments** | The issue was easily fixed by creating a method in the House class to add MovieSession to its List<MovieSession>. Create the list if not yet created before adding any MovieSession. |

**Report 4**

|  |  |
| --- | --- |
| **Title** | NullPointerException: Select Movie Module |
| **Date** | 3 November 2024 |
| **System Information** | Operating System: Microsoft Windows 11 Home  OS build: 10.0.22631 Build 22631  JDK version: 21 |
| **Severity** | Moderate |
| **Reported By** | Jasper |
| **Status** | Resolved |
| **Problem** | Exception in thread "main" java.lang.NullPointerException: Cannot invoke "selectMovieModule.MovieSession.getMovie()" because the return value of "java.util.Map.get(Object)" is null  at selectMovieModule.ToStringUtil.showMovieSessions(ToStringUtil.java:55)  at selectMovieModule.InputMenu.buyTicketMenu(InputMenu.java:165)  at selectMovieModule.InputMenu.customerMenu(InputMenu.java:131)  at selectMovieModule.InputMenu.guestStartMenu(InputMenu.java:80)  at selectMovieModule.CinemaSystem.main(CinemaSystem.java:19)  Encountered the above exception when trying to get the movie from MovieSession object. |
| **Steps to reproduce** | Invoke the function that return the string presentation of MovieSession objects |
| **Expected output** | Should return the name of the movie from the first MovieSession object |
| **Actual output** | Received a NullPointerException. |
| **Comments** | Forgot that I was using the data structure of Map<Integer, MovieSession> to hold the MovieSession objects instead of array so the first object does not necessarily start from index 0. As the first object’s key was 1, by simply changing get(0) to get(1), the issue was resolved. |

**Report 5**

|  |  |
| --- | --- |
| **Title** | Incorrect checking mechanism for valid input |
| **Date** | 3 November 2024 |
| **System Information** | Operating System: Microsoft Windows 11 Home  OS build: 10.0.22631 Build 22631  JDK version: 21 |
| **Severity** | Moderate |
| **Reported By** | Jasper |
| **Status** | Resolved |
| **Problem** | A screenshot of a computer program  Description automatically generated  When selecting the seats from the seating plan, even when the seats were available e.g. b12, g10, etc., the program still complained that the format was invalid |
| **Steps to reproduce** | 1. Load default data to database  2. Create a scheduleManagement object  3. Schedule the movies based on the default schedule algorithm  4. Auto log in as a customer  5. Choose the select movie option  6. Proceed till the step of seat selection |
| **Expected output** | Should be able to book the seats if shown to be available and display a message of successfully booking the selected seat |
| **Actual output** | Showed invalid input format for the selected seats |
| **Comments** | It was actually a careless and minor mistake in designing the checking mechanism for the input data for seats. Since the input can either be 2 or 3 characters e.g. A1, B12, the original checking mechanism checks if the input is only 2 characters which does not cover all cases. By simply changing the code to check for 2 or 3 characters, the problem was resolved. |

**Report 6**

|  |  |
| --- | --- |
| **Title** | Incorrect String Output Format for Movie Information |
| **Date** | 3 November 2024 |
| **System Information** | Operating System: Microsoft Windows 11 Home  OS build: 10.0.22631 Build 22631  JDK version: 21 |
| **Severity** | Minor |
| **Reported By** | Jasper |
| **Status** | Resolved |
| **Problem** | A close-up of a white background  Description automatically generated  For the column, ‘Scheduled?’, it should show either ‘Yes’ or ‘No’ on the same column. While the information was presented correctly, it went to the next line, which could affect users’ understanding and impact aesthetic value. |
| **Steps to reproduce** | 1. Load default data to database  2. Create a scheduleManagement object  3. Schedule the movies based on the default schedule algorithm  4. Auto log in as admin  5. Select the option of show all movies |
| **Expected output** | The results of the ‘Scheduled’ column should display under the same column and within the same row as the rest of the data of the same movie but not on a new line. |
| **Actual output** | The results of the ‘Scheduled’ column were displayed on a new line and did not match with the column. |
| **Comments** | The function that returns the string information of each movie had unnecessarily added a line break i.e. “\n”, as shown below:  A screen shot of a computer program  Description automatically generated  By simply removing the line break when building the string data, the issue was quickly resolved. |

**Report 7**

|  |  |
| --- | --- |
| **Title** | NullPointerException in Database Class |
| **Date** | 12 November 2024 |
| **System Information** | Operating System: Microsoft Windows 11 Home  OS build: 10.0.22631 Build 22631  JDK version: 21 |
| **Severity** | Minor |
| **Reported By** | Jasper |
| **Status** | Resolved |
| **Problem** | Exception in thread "main" java.lang.NullPointerException: Cannot invoke "java.util.List.isEmpty()" because "this.movies" is null  at release.database.Database.setMovies(Database.java:139)  at release.database.DefaultData.loadDefaultMoviesToDB(DefaultData.java:68)  at release.Main.main(Main.java:17) |
| **Steps to reproduce** | Create a singleton Database object  Invoke the function setMovies() |
| **Expected output** | Should successfully set the Database movies with the input parameter |
| **Actual output** | Induced the NullPointerException when checking List<Movies> in the Database Class |
| **Comments** | Didn’t know that the function, isEmpty(), cannot be applied to uninitialized List (I thought uninitialized/ null List was still considered an empty list). Thus, should add an additional checking to check if the List is null first before invoking other List supported functions. |

**Report 8**

|  |  |
| --- | --- |
| **Title** | UnsupportedOperationException in Database class |
| **Date** | 13 November 2024 |
| **System Information** | Operating System: Microsoft Windows 11 Home  OS build: 10.0.22631 Build 22631  JDK version: 21 |
| **Severity** | Minor |
| **Reported By** | Jasper |
| **Status** | Resolved |
| **Problem** | Encountered the issue of not being able to add newly created ProductWithPortion objects to database. Used the debugger to find out that an UnsupportedOperationException had been thrown. |
| **Steps to reproduce** | Load default data to database, including List<ProductWithPortion>  Create new ProductWithPortion objects and add them to database with the addProduct() function |
| **Expected output** | Should successfully add new ProductWithPortion objects to database |
| **Actual output** | Fail to add with the exception java.lang. UnsupportedOperationException thrown |
| **Comments** | The issue was due to creation of default List<ProductWithPortion> with Arrays.asList, which I didn’t know was immutable, thus making the list unable to handle the addition of new object. By simply using new ArrayList<>(Arrays.asList(xxx)), the resulting list will be mutable. |

**Report 9**

|  |  |
| --- | --- |
| **Title** | No PaymentRecords could be shown |
| **Date** | 14 November 2024 |
| **System Information** | Operating System: Microsoft Windows 11 Home  OS build: 10.0.22631 Build 22631  JDK version: 21 |
| **Severity** | Severe |
| **Reported By** | Jasper |
| **Status** | Resolved |
| **Problem** | Even with successful payment for products, no payment records could be shown. |
| **Steps to reproduce** | 1. Buy movie tickets 2. Keep proceeding till the payment step 3. Achieve the state of successful payment 4. List previous payment records |
| **Expected output** | For any successful payment, its record should be recorded. And when selecting the option of listing payment records, it should actually list out all previous records |
| **Actual output** | No records could be shown despite successful payment. |
| **Comments** | The issue lies in the function of clearing the shopping cart after successful payment. Since both payment record and shopping cart may share some common objects e.g. list of movie tickets, snacks/drinks, if directly set the payment record with the list of objects from shopping cart instead of creating a new list specifically for payment record, the list is actually shared by both payment record and shopping cart, so the clear function of shopping cart will also clear the same list shared by payment record. To resolve this, simply create a new list for payment record but not to set it as a pointer reference to the shopping cart list. |

**Report 10**

|  |  |
| --- | --- |
| **Title** | Search user function always shows all users in any searches |
| **Date** | 16 November 2024 |
| **System Information** | Operating System: Microsoft Windows 11 Home  OS build: 10.0.22631 Build 22631  JDK version: 21 |
| **Severity** | Minor |
| **Reported By** | Jasper |
| **Status** | Resolved |
| **Problem** | The search user function always shows all the users currently registered in the database no matter what search keyword is entered. |
| **Steps to reproduce** | 1. Log in as admin 2. Select the search user function 3. Enter any keyword to search |
| **Expected output** | It should only show the user with either its ID, name, or username that matches the search keyword. |
| **Actual output** | It lists out every user currently registered in the database. |
| **Comments** | It was a minor error in one line of the code. Instead of looping for the database for all users, it should actually loop through the newly created array that holds the found users from the database, as shown below: |

**Report 11**

|  |  |
| --- | --- |
| **Title** | Failure to handle values beyond Integer. MAX\_VALUE and Integer. MIN\_VALUE |
| **Date** | 19 November 2024 |
| **System Information** | Operating System: Windows 11 Home 23H2  OS build: 22631.4317  JDK version: 23 |
| **Severity** | Minor |
| **Reported By** | Thomas |
| **Status** | Resolved |
| **Problem** | When using the method Integer.parseInt() to handle input values, an exception that was not within our development preparation was caught, showing a weird message e.g. For input string: “2147483648” |
| **Steps to reproduce** | 1. Run the jar program 2. When asked for selecting an option, input any value that is greater than java’s Integer. MAX\_VALUE i.e 2147483647 or less than Integer.MIN\_VALUE i.e. -2147483648 |
| **Expected output** | It should only show an exception message defined by us e.g. no such option, but not an unexpected exception message. |
| **Actual output** | It shows an exception message, For input string: “2147483648”. |
| **Comments** | In Java, integer has a limit on its maximum and minimum value. Beyond that, the built-in method, Integer.parseInt(), will not be able to handle the value and will instead throw a NumberFormatException. By simply rewriting a small part of our code to catch the more specific NumberFormatException in the catch block, and once caught, we define our own exception message to be printed out, this issue is quickly resolved. |

**Report 12**

|  |  |
| --- | --- |
| **Title** | Cannot select payment record again after inputting some characters that cannot be converted to integer |
| **Date** | 26 November 2024 |
| **System Information** | Operating System: Windows 11 Home 23H2  OS build: 22631.4317  JDK version: 23 |
| **Severity** | Low |
| **Reported By** | Thomas |
| **Status** | Resolved |
| **Problem** | I type some characters instead of numbers into the console when choosing a payment method, and it asks me to select seats instead of asking me to try again. |
| **Steps to reproduce** | Login as a Member  Enter “Buy Movie Tickets” or 5.  Choose a movie session  Select seat  Type 2 to proceed to payment  Type some characters like “a” |
| **Expected output** | It should print out “[Exception] Selected payment option is invalid” and ask the member to select a payment method again. |
| **Actual output** | It prints out “[Exception] Invalid input format” and the seating plan. Then, it asks the member to select their seats again. |
| **Comments** | It turns out that the try block does not cover the Integer.parseInt method, so when the input value cannot be converted into an integer, the method for making payment (CmdDoPayment.execute) cannot catch the exception. Still, the method for choosing seats (CmdBuyMovieTickets.selectSeatMenu) catches it, so it proceeds to choose the seating plan again. It can be fixed by moving the Integer.parseInt method into the try block to handle the exception when the value cannot be converted into an integer, as demonstrated by the following picture. |

**Report 13**

|  |  |
| --- | --- |
| **Title** | Cannot continue to add snacks/drinks after attempting to add them with insufficient arguments. |
| **Date** | 27 November 2024 |
| **System Information** | Operating System: Windows 11 Home 23H2  OS build: 22631.4317  JDK version: 23 |
| **Severity** | Low |
| **Reported By** | Thomas |
| **Status** | Resolved |
| **Problem** | When I try to add the snacks with insufficient arguments, the system will return to the main menu directly instead of asking me to insert the snacks again.  It also happens when I attempt to add drinks. |
| **Steps to reproduce** | 1. Login as an admin 2. Enter “Add Snacks/Drinks” or 12. 3. Choose Snack (1) or Drink (2) 4. Enter the new snacks/drinks with insufficient arguments (e.g. “a,b”, not “a,b,1”, which is a valid input) |
| **Expected output** | It should ask the admin to add the new snacks/drinks again. |
| **Actual output** | It prints out “[Exception] 3 values must be provided” and returns to the main menu. |
| **Comments** | It turns out that the exception ExInsufficientArgs is not handled using the method for adding snacks and drinks (CmdAddSnacksDrinks.execute). It is thrown to the upper call stack and handled by the main method (Main.main). Therefore, the main menu is displayed instead of asking the admin to add a snack/drink again.    It can be fixed easily by handling the exception ExInsufficientArgs inside CmdAddSnacksDrinks.execute, instead of throwing it out to upper call stack directly. |

**Report 14**

|  |  |
| --- | --- |
| **Title** | Add member test case conflict with other test case. |
| **Date** | 28 November 2024 |
| **System Information** | Operating System: Windows 11 Home 24H2  OS build: 26100.2454  JDK version: 21 |
| **Severity** | Low |
| **Reported By** | Alan |
| **Status** | Resolved |
| **Problem** | The first testAddMember\_true function is smooth.    The second return the wrong result. |
| **Steps to reproduce** | 1. Run the whole test file. |
| **Expected output** | All test cases present true. |
| **Actual output** | The second add member function return false. |
| **Comments** | The problem is because the newly added user of the previous function has not been deleted. Based on the design, usernames should be unique; for this reason, the second test case returns wrong result due to the same username being used for testing, that username already occupied by another user who should not exist.  It was fixed by utilizing the delUser function of UserCenter after the assertion, to delete the added user for testing purposes only. |