**City University of Hong Kong**

2017-2018 Semester A

CS3343 Software Engineering Practice

**Bug Report**

Maverick:

The Smart Trading Platform

Group 14

|  |  |
| --- | --- |
| **Name** | **Student Number** |
| Dheeraj Khurana | 54227224 |
| Navwinder Singh | 54395023 |
| Ravinder Singh | 54390586 |
| Pang TSZ HO | 54387730 |
| Mathew Bilo | 54387858 |

**Bug 01** – **Program stopped working on unknown category**

|  |  |  |  |
| --- | --- | --- | --- |
| **Status:** | VERIFIED FIXED | **Reporter:** | Matthew Bilo |
| **Product:** | Smart Trading Platform | **Assignee:** | Singh Ravinder |
| **Component:** | cmdBuyProduct | **CC List:** | Dheeraj Khurana  Navwinder Singh  Pang Tsz Ho |
| **Version:** | 1.85 |
| **Hardware:** | PC |  |  |
| **OS:** | Windows 10 |  |  |
| **Severity:** | major |  |  |

Matthew Bilo

Steps To Reproduce:

1. Login
2. Choose Buy
3. Input a category not defined in the program

Actual Results:

Program stop working

Expected Results:

Program ask user to input a correct category

Singh Ravinder

Added handling for unknown category in cmdBuyProduct

**if** (productsList.size() == 0) {

System.*out*.println("Please select one of the above types only, otherwise state the Others category.");

}

**Bug 02** – **Program stopped working on unknown category**

|  |  |  |  |
| --- | --- | --- | --- |
| **Status:** | VERIFIED FIXED | **Reporter:** | Pang Tsz Ho |
| **Product:** | Smart Trading Platform | **Assignee:** | Matthew Bilo |
| **Component:** | cmdBuyProduct | **CC List:** | Dheeraj Khurana  Navwinder Singh  Singh Ravinder |
| **Version:** | 1.85 |
| **Hardware:** | PC |  |  |
| **OS:** | Windows 10 |  |  |
| **Severity:** | major |  |  |

Pang Tsz Ho

Steps To Reproduce:

1. Login
2. Choose Sell
3. Input a category not defined in the program

Actual Results:

Program stop working

Expected Results:

Program ask user to input a correct category

Matthew Bilo

Added handling for unknown category in cmdSellProduct

**do**{

System.*out*.println("Please enter the category/type of the product");

category = in.nextLine();

**if** (!category.equals("Electronic Appliances") && !category.equals("Furniture") && !category.equals("Grocery") && !category.equals("Apparel") && !category.equals("Books") && !category.equals("Sports") && !category.equals("Others")){

System.*out*.println("Invalid Input");

System.*out*.println(category);

}

}**while**(!category.equals("Electronic Appliances") && !category.equals("Furniture") && !category.equals("Grocery") && !category.equals("Apparel") && !category.equals("Books") && !category.equals("Sports") && !category.equals("Others"));

**Bug 03** – **Program stopped working on mismatch product/seller**

|  |  |  |  |
| --- | --- | --- | --- |
| **Status:** | VERIFIED FIXED | **Reporter:** | Dheeraj Khurana |
| **Product:** | Smart Trading Platform | **Assignee:** | Navwinder Singh |
| **Component:** | cmdBuyProduct | **CC List:** | Pang Tsz Ho  Matthew Bilo  Singh Ravinder |
| **Version:** | 1.86 |
| **Hardware:** | PC |  |  |
| **OS:** | Windows 10 |  |  |
| **Severity:** | major |  |  |

Dheeraj Khurana

Steps To Reproduce:

1. Login
2. Choose Buy
3. Input a product/seller not not listed in the product list

Actual Results:

Program stop working

Expected Results:

Program ask user to input a correct product/seller

Navwinder Singh

Fixed mismatch product/seller problem in cmdBuyProduct

**if** (purchase == **null**){

System.*out*.println("\nProduct Name and Seller Name does not match. Please try again!");

}

**Bug 04** – **Program stopped working on non-integer payment**

|  |  |  |  |
| --- | --- | --- | --- |
| **Status:** | VERIFIED FIXED | **Reporter:** | Singh Ravinder |
| **Product:** | Smart Trading Platform | **Assignee:** | Dheeraj Khurana |
| **Component:** | cmdBuyProduct | **CC List:** | Pang Tsz Ho  Matthew Bilo  Navwinder Singh |
| **Version:** | 1.86 |
| **Hardware:** | PC |  |  |
| **OS:** | Windows 10 |  |  |
| **Severity:** | normal |  |  |

Singh Ravinder

Steps To Reproduce:

1. Login
2. Choose Buy
3. Input a product and seller
4. Input a non-integer payment

Actual Results:

Program stop working

Expected Results:

Program ask user to input a integer

Dheeraj Khurana

non-integer payment in cmdBuyProduct is now handled

**boolean** isInt = **true**;

**do**{

isInt = **true**;

**try**{

Payment = Integer.*parseInt*(in.nextLine());

}**catch** (NumberFormatException ex) {

System.*out*.println("Please input an integer");

isInt = **false**;

}

}**while**(isInt == **false**);

**Bug 05** – **Program stopped working on non-integer price**

|  |  |  |  |
| --- | --- | --- | --- |
| **Status:** | VERIFIED FIXED | **Reporter:** | Singh Ravinder |
| **Product:** | Smart Trading Platform | **Assignee:** | Dheeraj Khurana |
| **Component:** | cmdSellProduct | **CC List:** | Pang Tsz Ho  Matthew Bilo  Navwinder Singh |
| **Version:** | 1.86 |
| **Hardware:** | PC |  |  |
| **OS:** | Windows 10 |  |  |
| **Severity:** | normal |  |  |

Singh Ravinder

Steps To Reproduce:

1. Login
2. Choose Sell
3. Input a product name
4. Input a non-integer price

Actual Results:

Program stop working

Expected Results:

Program ask user to input a integer

Dheeraj Khurana

non-integer payment in cmdSellProduct is now handled

**boolean** isInt = **true**;

**do**{

isInt = **true**;

**try**{

Payment = Integer.*parseInt*(in.nextLine());

}**catch** (NumberFormatException ex) {

System.*out*.println("Please input an integer");

isInt = **false**;

}

}**while**(isInt == **false**);

**Bug 06** – **Date computation not working**

|  |  |  |  |
| --- | --- | --- | --- |
| **Status:** | VERIFIED FIXED | **Reporter:** | Navwinder Singh |
| **Product:** | Smart Trading Platform | **Assignee:** | Pang Tsz Ho |
| **Component:** | Discount, Shipping, RewardPoints | **CC List:** | Matthew Bilo  Singh Ravinder  Dheeraj Khurana |
| **Version:** | 1.90 |
| **Hardware:** | PC |  |  |
| **OS:** | Windows 10 |  |  |
| **Severity:** | major |  |  |

Navwinder Singh

Test cases related to date computation failed. E.g. testBirthdayDiscount\_02()

Steps To Reproduce:

1. Run testBirthdayDiscount\_02()

Actual Results:

Function return 1

Expected Results:

Function return 0.85

Pang Tsz Ho

Did not initialize the Calendar object

Before:

**private** Calendar todayDate;

After:

**private** Calendar todayDate = Calendar.*getInstance*();

Static field DAY\_OF\_MONTH and MONTH are being access in non static ways

Before:

**public** **double** BirthdayDiscount(User user)

{

**if** (todayDate.DAY\_OF\_MONTH != user.getBirthdayDay() && todayDate.MONTH == user.getBirthdayMonth())

{

**return** 0.1;

}

**else** **if** (todayDate.DAY\_OF\_MONTH == user.getBirthdayDay() && todayDate.MONTH == user.getBirthdayMonth())

{

**return** 0.15;

}

**else** **return** 0;

}

After:

**public** **double** BirthdayDiscount(User user, Calendar date)

{

**if** (date.get(Calendar.*DAY\_OF\_MONTH*) != user.getBirthdayDay() && date.get(Calendar.*MONTH*) == user.getBirthdayMonth())

{

**return** 0.9;

}

**else** **if** (date.get(Calendar.*DAY\_OF\_MONTH*) == user.getBirthdayDay() && date.get(Calendar.*MONTH*) == user.getBirthdayMonth())

{

**return** 0.85;

}

**else** **return** 1;

}

**Bug 07** – **Wrong Discount are computed**

|  |  |  |  |
| --- | --- | --- | --- |
| **Status:** | VERIFIED FIXED | **Reporter:** | Dheeraj Khurana |
| **Product:** | Smart Trading Platform | **Assignee:** | Pang Tsz Ho |
| **Component:** | cmdSellProduct | **CC List:** | Matthew Bilo  Navwinder Singh  Singh Ravinder |
| **Version:** | 1.91 |
| **Hardware:** | PC |  |  |
| **OS:** | Windows 10 |  |  |
| **Severity:** | major |  |  |

Dheeraj Khurana

Test cases computing discount failed. E.g. testMaxDiscount\_02()

Steps To Reproduce:

1. Run testMaxDiscount\_02()

Actual Results:

Function return 0.2 + 0.15

Expected Results:

Function return 0.8 \* 0.85

Pang Tsz Ho

Discount is implemented in a wrong way e.g. 70% off should be 0.3 instead of 0.7. Also the computation of two discount should be multiplication instead of addition.

Before:

**public** **double** MaxDiscount()

{

**double** total = 0;

**double**[] discounts = {basicDiscount, groceryDiscount, premiumDiscount, birthdayDiscount, seasonalDiscount};

Arrays.sort(discounts);

total = (discounts[discounts.length-1]) + (discounts[discounts.length-2]);

**if** (total >= 0.3)

{

total = 0.3;

}

**return** total;

}

After:

**public** **double** MaxDiscount()

{

**double** total = 0;

**double**[] discounts = {basicDiscount, groceryDiscount, premiumDiscount, birthdayDiscount, seasonalDiscount};

Arrays.*sort*(discounts);

total = (discounts[0]) \* (discounts[1]);

**if** (total <= 0.3)

{

total = 0.3;

}

**return** total;

}