**Module Code & Module Title:**

**CS5004NT Emerging Programming Platforms and Technologies**

**Assessment Weightage & Type:**

**30% Group Coursework**

**Title:**

**Smartphone Accessory details**

**Year and Semester:**

**2020-21 Autumn**

| Group Name: (C4G6) | | | |
|---|---|---|---|
| SN | Student Name | College ID | University ID |
| 1 | Ayush Mahatara | NP05CP4A190037 | 19031611 |
| 2 | Delish khadka | NP05CP4A190051 | 19031630 |
| 3 | Nischal Rai | NP05CP4A190096 | 19031772 |
| 4 | Manzil Katwal | NP05CP4A190083 | 19031758 |

*I confirm that I understand my coursework needs to be submitted online via Google Classroom under the relevant module page before the deadline in order for my assignment to be accepted and marked. I am fully aware that late submissions will be treated as non-submission and a mark of zero will be awarded.*

# Acknowledgement

The report about developing an Inventory Information System was created with the guidance of our seniors and our module leader Mr. Pradhumna Dhungana.

Many persons have generously shared their ideas about report and helped us by pointing out some errors and helping how to deal with it in a proper way. We are grateful for all their suggestions. They helped us to write more clearly and accurately, which made this report more meaningful. We express our sincere appreciation to our Module leader Mr. Pradhumna Dhungana for all the useful guidance. We are thankful to Itahari International College and London Metropolitan University for providing us such a useful module to work on. We express our sincere appreciation to our useful mentors for their suggestions, constructive criticisms, and encouragement.

# Abstract

The creative process in which we instruct the computer and other electronic devices to how to perform the specific task is known as programming and the language which is used to program the computer is known as programming language. An organizational system designed to collect, process, store, and distribute information is known as Information System.

London Metropolitan University has a module named “Emerging Programming Platforms and Technologies” in the 3rd semester for computing students which is a groupwork for the computing student, given to understand the emerging platform.

The report section consists of the development of and Information System name as “Optimus IS”. The working of Information System is shown in the user manual. Binary Search Algorithm is used to find the targeted value. Validation and testing are carried out to see the compilation of requirement of the Information System. Pseudocode is also provided in the report for the better understanding. We can find the proper description in the method description table. The task of the groupwork was equally divided among the group members and we also helped each other in some difficult parts of the reports.

## Table of Contents

Table of Figures

Table of Tables

## 1.Introduction

The creative process of designing and building an executable computer program that instruct a computer on how to do a specific task is known as programming (Bolton, 2019). The computer language that can be understood by computers and that is used by programmers to develop software programs, scripts, or other sets of instructions that computers can execute is called a programming language... (Computer Hope, 2020). Programming language is divided into two types: low-level programming language and high-level programming language. Assembly language and machine language are examples of low-level programming language. C++, Java, Python, PHP, etc. are some examples of high-level programming language.

One of the most used high-level programming languages, Java is used in this module. Java is an object-oriented Programming language, which is fast, reliable, and secure. Like other high-level programming language, Java has its own syntax and structure format for coding. Java Development Kit (JDK), Java Runtime Environment (JRE) and Java Virtual Machine (JVM) are three main components of programming environment of Java (Rishabh Prabhu, 2019).

In this module, NetBeans is used to develop the program. In NetBeans, we can create GUI by just dragging the available components, so NetBeans is commonly used by programmers to build GUI. Most used shapes in different sizes that are needed for GUI can be found in NetBeans. As the NetBeans helps to use the syntax of Java programming language, it is highly recommended for the Java Programmers.

Optimus Smartphone Shop is a known organization, which have been dealing with selling smartphones to customers from Province 1 and Province 2. The proposed Information System was for providing the detail information about available smartphones to the user with suitable price. In this Information System, smartphone details are stored with the details like accessories number, accessories name, company name, price etc. The functions like search, insert, and delete can be seen, where user can fill the information the user will be able to go through the user manual provided with the help of menu bars. Menu bar designing with proper use of Menu, Menu Items, Table, Buttons. Combo.

Boxes, Radio Buttons, and Label that were used to enter the required details of the accessories can be visible in this Information System. Proper validation of every required item records done with proper error and success messages can also be found in this Information System. As the “Optimus IS” keep the record of all the available smartphones with needed detail, this Information System helps to save time and helps to increase the efficiency in daily task.

# 2.Proposal

## 2.1 Optimus Mobile Shop

Optimus Mobile Shop is the oldest and one of the best smartphones selling shop in Nepal. It is in Itahari, Province-1, Nepal. As Optimus Mobile Shop provides any kind of “Android” as well as “IOS” smartphones with best price, it is very popular in Province-1 and Province-2. Most of the people choose Optimus Mobile Shop as their first priority shop. The main reason of Optimus Mobile Shop to be one of the best smart phones selling is that it provides cheapest to advance smart phones that brings high customer satisfaction.

## 2.2 Situation

In this Modern Era of growing science and technology, smartphone is the most needed device among the people. With the growing population of the country, the demand and requirement for smartphone is increasing in the market. Due today by day growing requirement of the smartphone, the order, and the need of detail specification in the shop is increasing day by day. Optimus Mobile Shop is facing problem in handling the customer’s order, which may cause dissatisfaction in customers. Optimus Mobile Shop is also facing problems in recording the information about customers. Staff are also having problem in showing the past records of the customers. To solve these kinds of problem an Inventory Information System is needed by Optimus Mobile Shop.

## 2.3 Proposed Solution

Observing the problem faced by Optimus Mobile Shop and looking at the growing business of the organization an Inventory Information System known as “Optimus IS” will be proposed to Optimus Mobile Shop. This Inventory Information System will help to utilize the time for searching smartphones as the user requirement. The Inventory Information System will be developed in such way that it will be able to search details of the required smartphone in very short interval of time. This system will help the organization to increase the customer satisfaction and also to give customers detail information about the available smartphones and parts in the organization at the best price.

## 2.4 List of data

1. Model Number: The accessories are identified by numbers. Each accessory will be assigned with unique numbers so they can be accessed easily. This data will help us to process information about an accessory of specific model number

2. Accessory Name: Accessory name will contain name of phone parts like display, battery etc. or even new phone.

3. Category: It will contain different types of categories of accessories. Here, the categories will be Apple, Samsung, Huawei etc.

4. Recommendation: It will contain the accessories recommendation which will show who have recommended the accessories for the best deal.

5. Quality: It will represent the quality of the accessories i.e. Low, Medium and High

6. Price: It will contain the price of the accessories.

## 2.5 Steps Involved

- “Optimus IS” is will be introduced to the organization.
- Staffs will be familiarized and taught the methods about how to use this Information System.
- This IS will be used daily for maintaining all types of records related to smartphone which helps the organization in future references.
- If any problem arises between customers, the IS will be utilized to overcome such kind of problems.

## 2.6 Benefits

✓ This Information System will bring high customers satisfaction because it will take less time to provide detailed information about the smartphones as per the customers requirement.

✓ As the Information System provides information in less time, we can handle more customers than before.

✓ The high number of manpower isn’t required because of the efficient work of the Information System.

✓ The users will be able to insert, retrieve and delete the data.

✓ The users will be able to search the smartphones and parts according to its price.

## 2.7 Potential Obstacles

- The Information System is expensive (approximately $1000).
- It nearly takes 4 weeks to include all the feature in “Optimus IS”.
- User must learn the basic about the “Optimus IS” before using it.
- Illiterate user won’t be able to use this type of Information System.

## 2.8 Tools

NetBeans- NetBeans is used for the development of the system. NetBeans is easier to use and easier to create GUI by dragging components available in required shape and size. It is efficient to use because of its editors, code analyzers, and converters. We can quickly and smoothly upgrade your applications to use new Java 8 language constructs, such as lambdas, functional operations, and method references in NetBeans. NetBeans is used for creating, testing, debugging, deploying and profile application.

Snipping tool: It is used for taking screenshot of the tests done. It helps us to take screenshot of screen.

**Individual Tasks**

Ayush Mahatara

- Method description
- GUI design
- Validation of code

Delish Khadka

- Pseudo code
- Binary search algorithm & flowchart
- Search by price button functionality

Nischal Rai

- Introduction
- Abstract

- Acknowledgement
- Menu Items
- proposal

Manzil Katwal

- Testing
- Search by category functionality
- Formatting
- Conclusion

# 3.Searching

The process of checking or retrieving elements from any data structure where it is stored is known as searching. Based on search type operation, algorithms searches are divided into two types: Linear Search Algorithm and Binary search Algorithm (Geeks for Geeks, 2020). Binary Search Algorithm use for creating Information System in this module.

## 3.1 Binary Search Algorithm

Binary Search Algorithm is one of the widely used searching techniques which can be used to sort arrays. The divide and conquer strategy searching technique followed by Binary Search Algorithm. The search space always reduces to half in every iteration (Parvez, 2020). Binary search is also known as half-interval search. Binary search is a search algorithm that finds the position of a target value within a sorted array and compares the target value to the middle element of the array.

In binary search at first, the target value to the middle element of the array is compared and if they are not equal, the half in which the target cannot lie is eliminated and the search continues on the other remaining half, again the middle element is taken to compare to the target value, and repeating this until the target value is founded.

**Example**

Item to be search = 20

Input:

| 0 | 1 | 2 | 3 | 4 |
|---|---|---|---|---|
| 10 | 11 | 16 | 20 | 23 |

The above table shows the sorted array and the item to be searched is 20. Here we have to find the index number where 20 lies. At first, we must determine the middle index.

**mid = (start + end) / 2**

Here,

beg = 0

end = 4

midIndex = (0+4)/2 = 4/2 = 2

beg = 0, end = 4, mid = 2

| 0 | 1 | 2 | 3 | 4 |
|---|---|---|---|---|
| 10 | 11 | 16 | 20 | 23 |

In the above table mid is 2 and here the value 20 is greater than 16 so it now checks only on the right half. So, it ignores the left half inducing the first midpoint assigned by it and keep 3 as beginning point.

| 2 | 3 | 4 |
|---|---|---|
| 16 | 20 | 23 |

Here,

beg = 2

end = 4

midIndex = (2+4)/2 = 6/2 = 3

beg = 2, end = 4, mid = 3

| 2 | 3 | 4 |
|---|---|---|
| 16 | 20 | 23 |

Element found at index 3, Hence 3 will get returned.

### 3.2 Algorithm of binary search

1. Start.

2. Take sorted arraylist with arguments.

3. Create start and end variables and value to be searched.

4. Initialize the value of start as 0.

5. Initialize the value of end size of arraylist size -1.

6. If the value of start is less or equals to end value, initialize the mid value as (end+start)/2

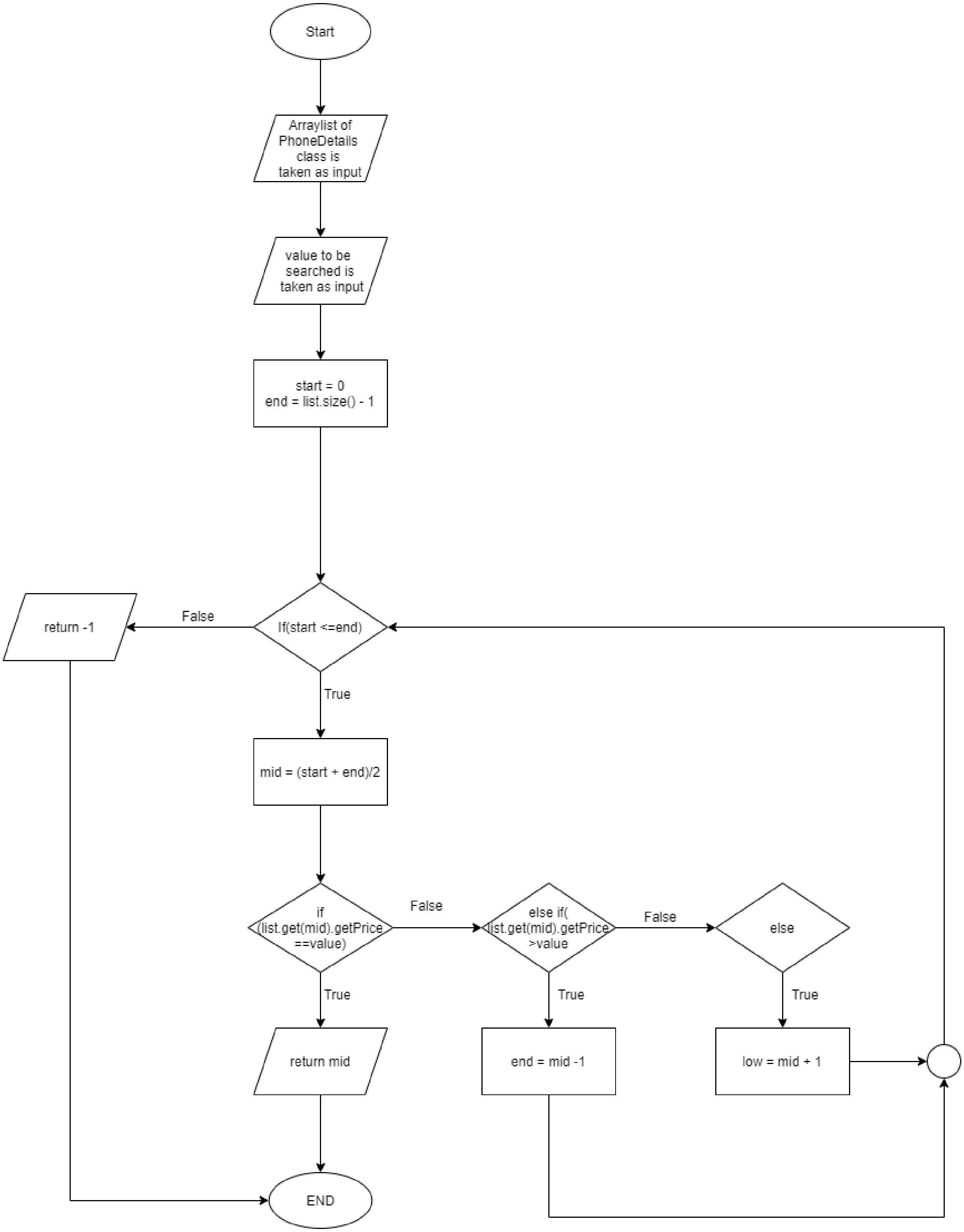
7. If the value of start is greater than or equal to end, return -1 and END.

8. If the price at the mid-index of the list is equal to ‘value,’ return mid and END.

9. Else if the price at mid- index is greater than value, initialize the value of end = mid – 1 and proceed to step 6.

10. Else, initialize value of low = mid + 1 and proceed to step 6.

### 3.3 Flowchart of binary search:


Start
Arraylist of PhoneDetails class is taken as input
value to be searched is taken as input
start = 0
end = list.size() - 1
If(start <=end)
False
return -1
True
mid = (start + end)/2
if (list.get(mid).getPrice ==value)
False
else if( list.get(mid).getPrice >value
False
else
True
return mid
True
end = mid -1
True
low = mid + 1
END

## 3.4 Pseudocode for binary search

```
DEFINE binarySearch of return type int(Call the Array List of PhoneDetails class, int value start from where the search begins, int end upto where the search is done, value which will be searched as parameter)
DO
IF (start is less or equals to end)
        INITIALIZE variable mid AND DO (end+start) divide by 2
        IF price at mid index is equals to value
                RETURN mid
        ELSE IF (price at mid index is less than value)
                RETURN  binarySearch of parameter list, mid + 1, end and value
        ELSE
                RETURN  binarySearch of parameter list, start, mid - 1 and value
        END IF
ELSE
        RETURN -1
END ELSE
END DO
```

## 4. Method discription

| Modifier and Type | Method and Description |
|---|---|
| void | addBtnActionPerformed<br>This method is used in inserting the values in the ArrayList and after checking all the required validation it is inserted in the table. |
| boolean | Validation<br>This method checks if the entered value of module number and accessory name are same or not for arraylist. It makes sure that the value to be unique. |
| void | exitBtnActionPerformed<br>This method helps user to exit the program asking them if they want to exit the program for sure. |
| Void | clearBtnActionPerformed<br>This method is used to clear the data entered by the user. |
| Void | searchCategoryBtnActionPerformed<br>This method is used to search the value which are inserted into the table according to their category selected in the combobox. |
| Void | btnDeleteActionPerformed<br>This method is used for deleting the row form the table if it is available and selected. |
| void | JMenuOpenActionPerformed |

| | |
|---|---|
| Void | JMenuExitActionPerformed<br>This method helps user to exit the program asking them if they want to exit the program for sure. |
| void | JMenuHelpActionPerformed<br>In this method, a txt file is opened for helping the user about the use of the program. |
| Void | Tolist<br>This method is used in retrieving the data from the table and setting all the value in the arraylist. |
| Void | binarySearch<br>In this method, the binary search algorithm is used which takes the mid value of the arraylist and checks with the value of parameter. If the value from the parameter is less or more than the mid value binarySearch method is called. |
| Void | Sort<br>This method is used for sorting the array list in ascending order. |
| Void | searchbtnActionPerformed |
| Void | This method is used for searching the values in the table according to the price. |
| Void | minimumPosition<br>This method is used for returning the minimum position to sort method. |

# 5. Testing

## 5.1 Test Scenario A**: Run your program on NetBeans**

| | |
|---|---|
| **Objectives** | Display running program on NetBeans |
| **Action** | Run program on NetBeans |
| **Expected Result** | Program will run on NetBeans |
| **Actual Result** | Program successfully ran on NetBeans |
| **Conclusion** | Test was successful |

*Table 1: Run Program on NetBeans*

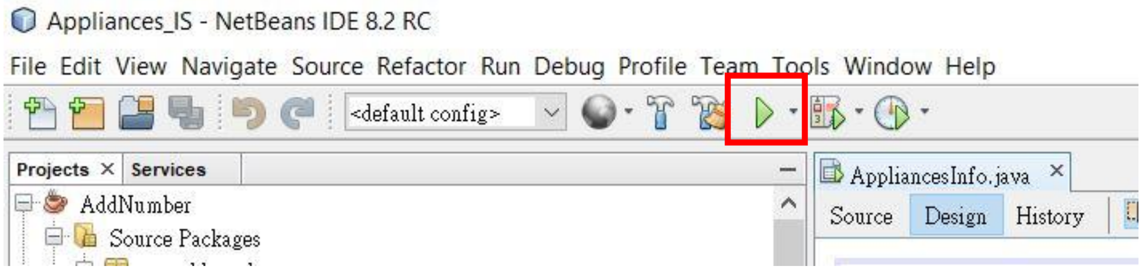
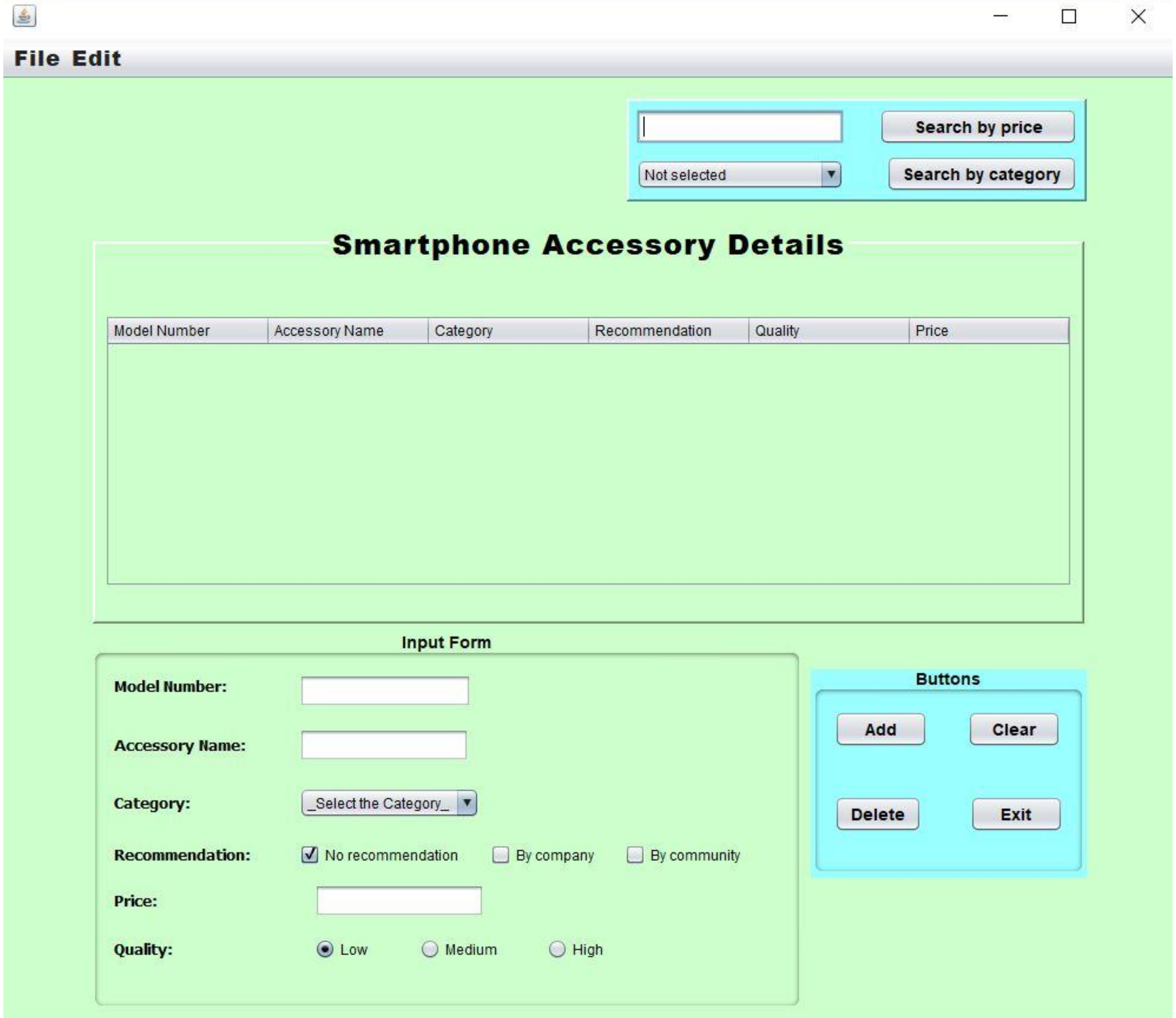

Appliances_IS - NetBeans IDE 8.2 RC
File Edit View Navigate Source Refactor Run Debug Profile Team Tools Window Help
<default config>
Projects
Services
AddNumber
Source Packages
AppliancesInfo.java
Source
Design
History

*Figure 1: Run Program*



| Model Number | Accessory Name | Category | Recommendation | Quality | Price |
|---|---|---|---|---|---|

Model Number:

Accessory Name:

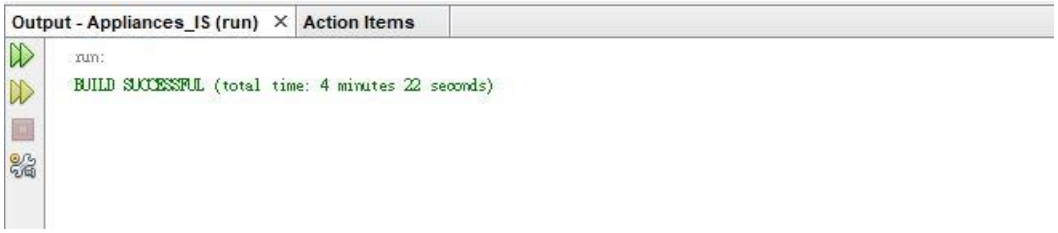
Category: _Select the Category_

Recommendation: ☑ No recommendation ☐ By company ☐ By community

Price:

Quality: ● Low ○ Medium ○ High

*Figure 2: Display GUI*



```
run:
BUILD SUCCESSFUL (total time: 4 minutes 22 seconds)
```

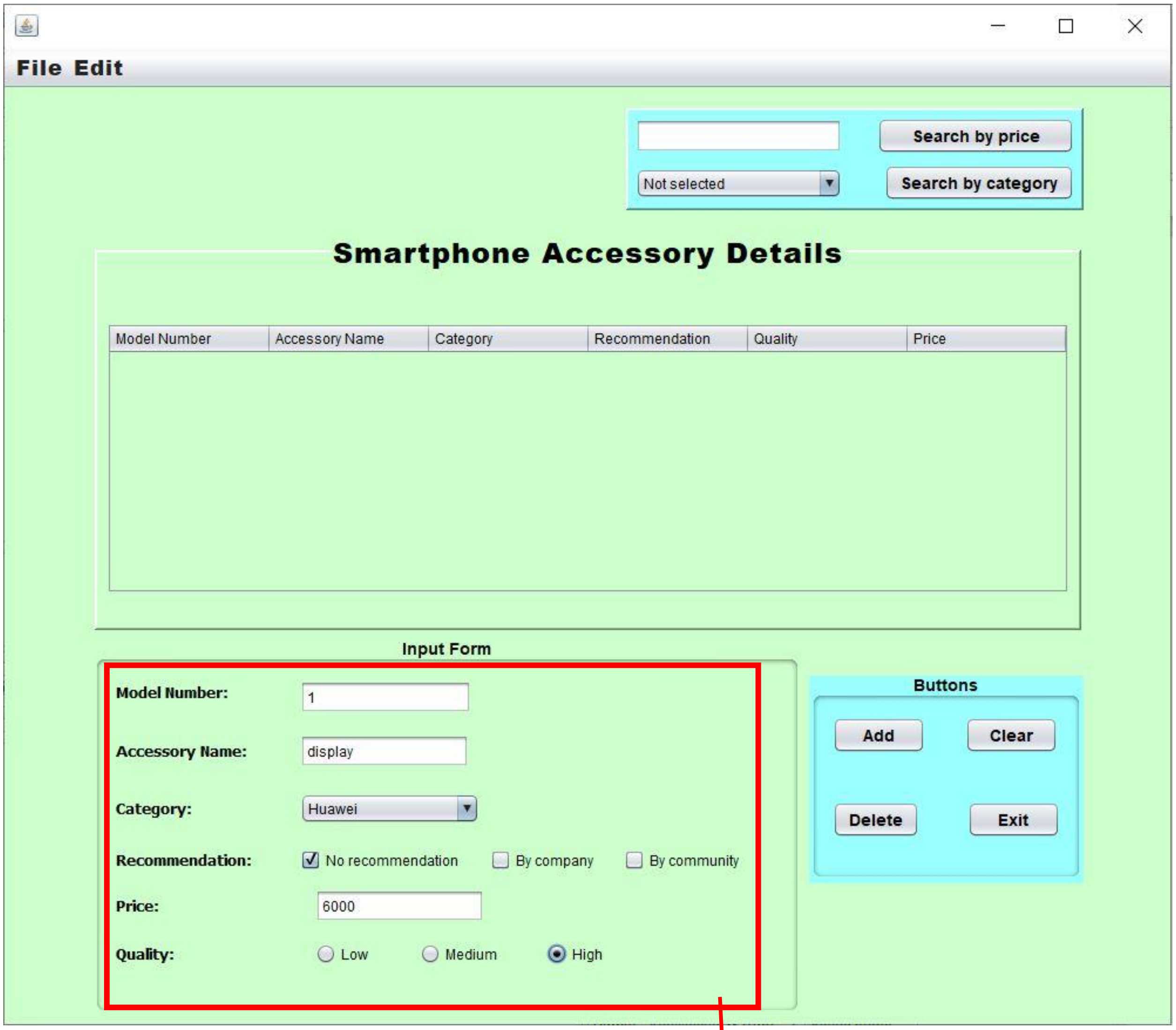
*Figure 3: Build Successful*

## 5.2 Testing Scenario B**: Functionality of the program**

### 5.2.1 Test B (1): Adding Item Details to table

| | |
|---|---|
| **Objectives** | Add items on table |
| **Action** | Insert value on text field<br>Model Number = “1”<br>Accessory Name = “Display”<br>Category = “Apple”<br>Recommendation =” No recommendation”<br>Price = 6000<br>Quality = low<br>Click Add Button |
| **Expected Result** | Values will be added on table |
| **Actual Result** | Values are added on table |
| **Conclusion** | Test was successful |

*Table 2: Testing Scenario B: Functionality of the program*



Not selected

Smartphone Accessory Details

| Model Number | Accessory Name | Category | Recommendation | Quality | Price |
|---|---|---|---|---|---|

Input Form

Model Number: 1

Accessory Name: display

Category: Huawei

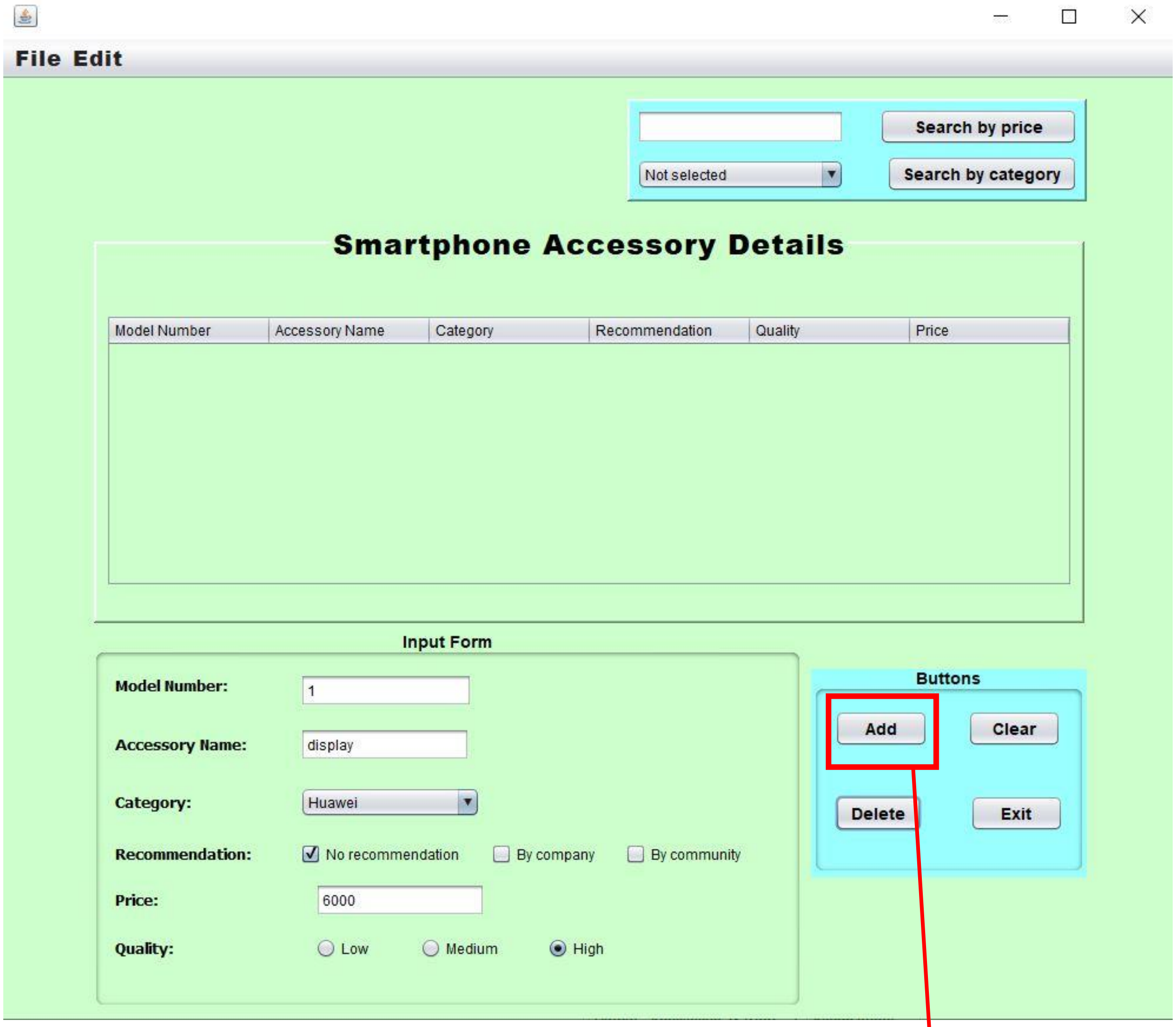
Recommendation: ☑ No recommendation ☐ By company ☐ By community

Price: 6000

Quality: ○ Low ○ Medium ● High

*Figure 4: Insert value in Text Field*

Inserting the values in the text fields



Not selected

| Model Number | Accessory Name | Category | Recommendation | Quality | Price |
|---|---|---|---|---|---|

Model Number: 1

Accessory Name: display

Category: Huawei

Recommendation: ☑ No recommendation □ By company □ By community

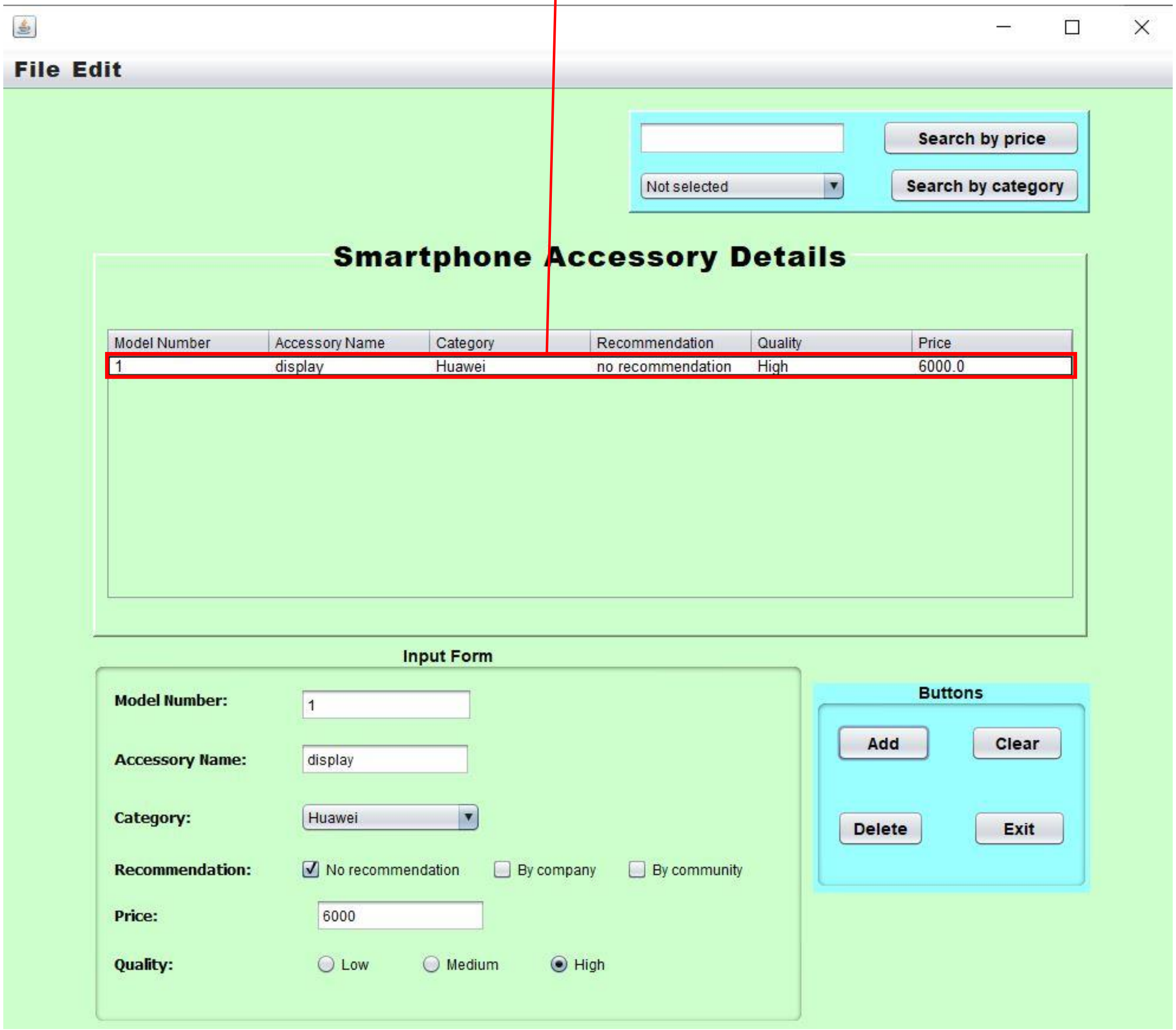
Price: 6000

Quality: ○ Low ○ Medium ● High

*Figure 5: Click add button*

Click add button

Values are added on the table



**Smartphone Accessory Details**

| Model Number | Accessory Name | Category | Recommendation | Quality | Price |
|---|---|---|---|---|---|
| 1 | display | Huawei | no recommendation | High | 6000.0 |

**Input Form**

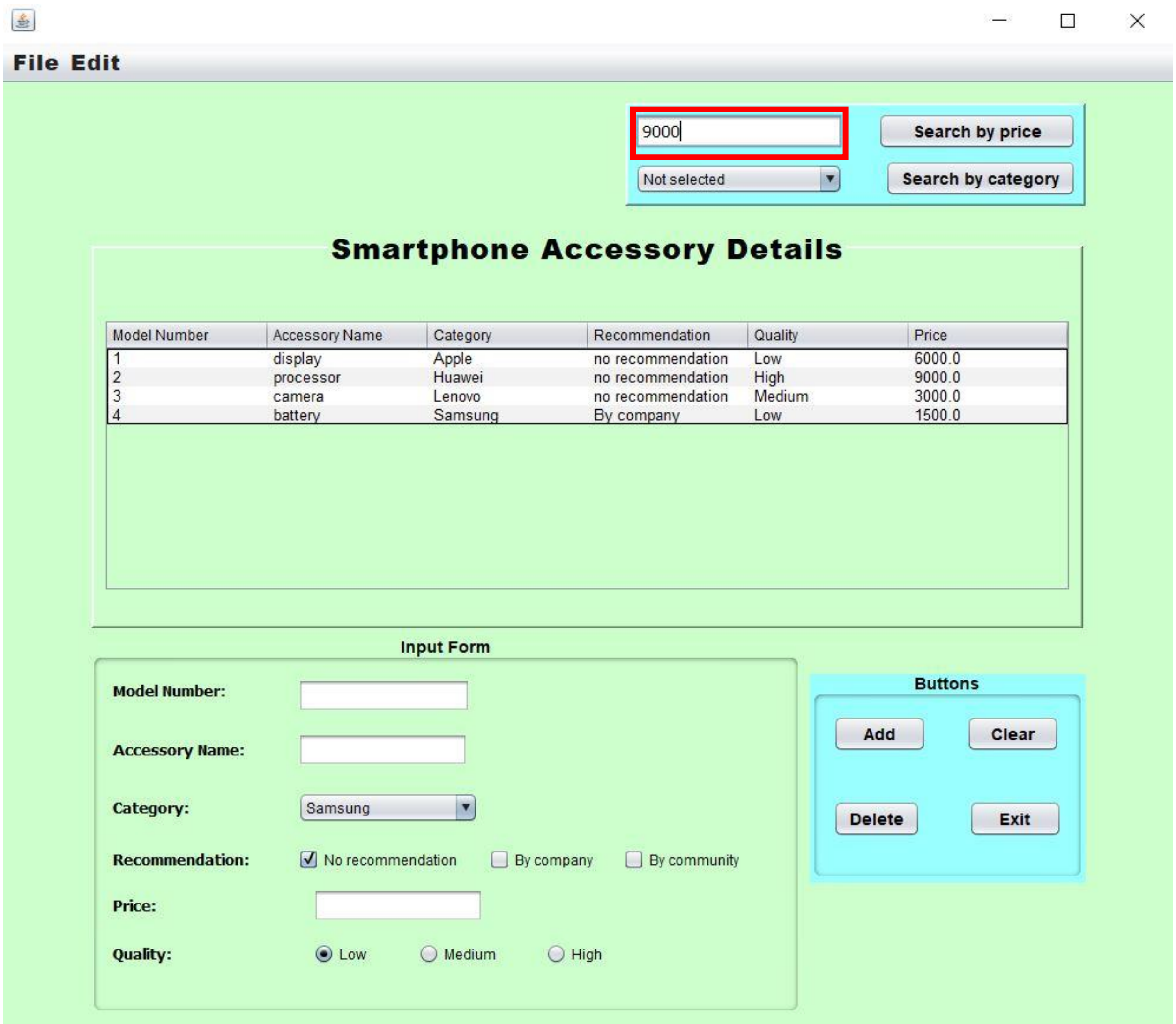
| | |
|---|---|
| Model Number: | 1 |
| Accessory Name: | display |
| Category: | Huawei |
| Recommendation: | ☑ No recommendation □ By company □ By community |
| Price: | 6000 |
| Quality: | ○ Low ○ Medium ● High |

*Figure 6: Add item on table*

### 5.2.2 Test B (2): searching for items based on price

| | |
|---|---|
| **Objectives** | Search item based on price |
| **Action** | Insert value on text field<br>Price = “9000”<br>Click on search button |
| **Expected Result** | Accessories Details will be shown in Dialog Box |
| **Actual Result** | Accessories Details is shown in Dialog Box |
| **Conclusion** | Test was successful |

*Table 3: Test B (2): searching for items based on price*



9000

Not selected

**Smartphone Accessory Details**

| Model Number | Accessory Name | Category | Recommendation | Quality | Price |
|---|---|---|---|---|---|
| 1 | display | Apple | no recommendation | Low | 6000.0 |
| 2 | processor | Huawei | no recommendation | High | 9000.0 |
| 3 | camera | Lenovo | no recommendation | Medium | 3000.0 |
| 4 | battery | Samsung | By company | Low | 1500.0 |

Model Number:

Accessory Name:

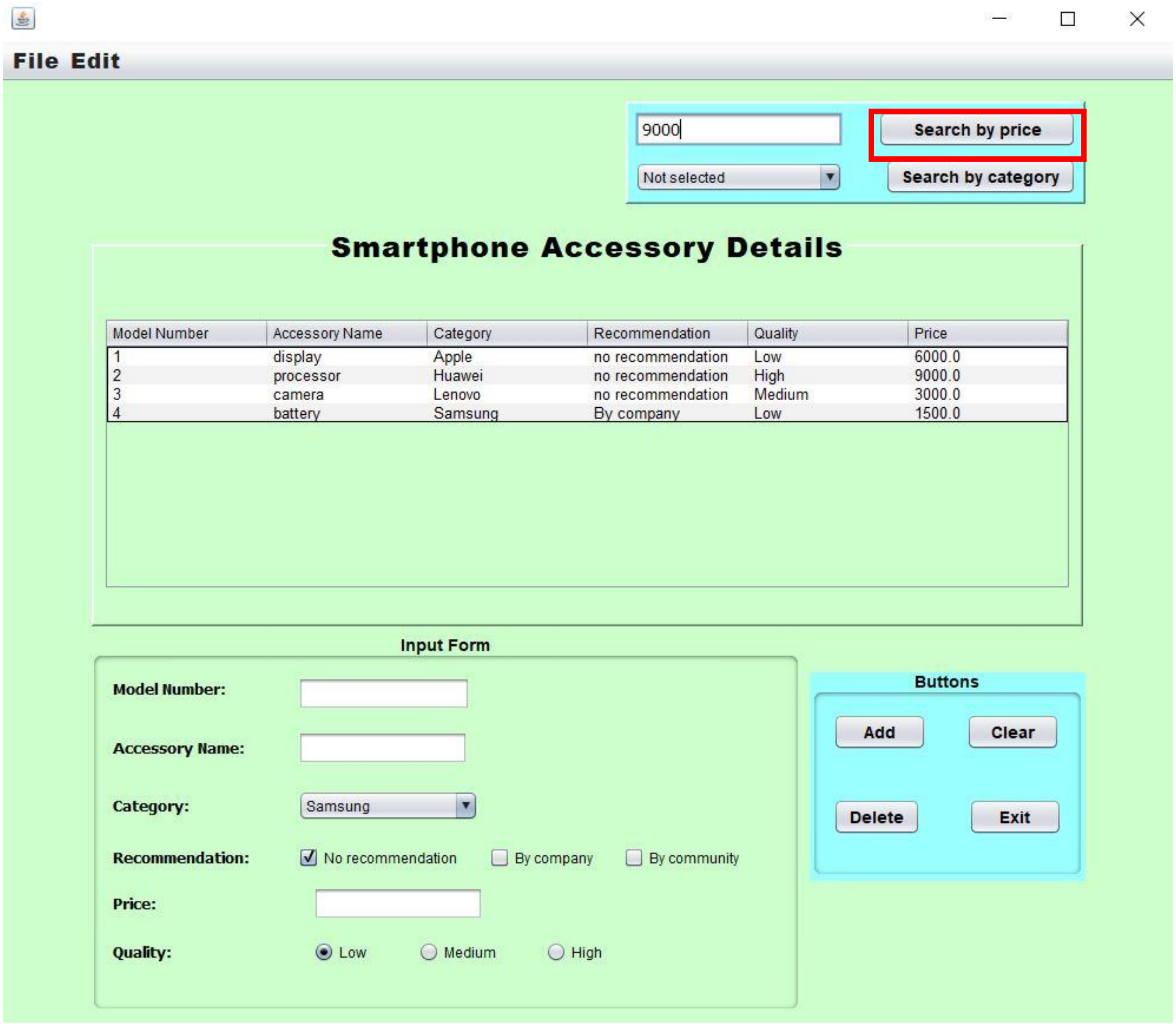
Category: Samsung

Recommendation: ☑ No recommendation ☐ By company ☐ By community

Price:

Quality: ● Low ○ Medium ○ High

*Figure 7: Insert value in text field*



9000

Not selected

Smartphone Accessory Details

| Model Number | Accessory Name | Category | Recommendation | Quality | Price |
|---|---|---|---|---|---|
| 1 | display | Apple | no recommendation | Low | 6000.0 |
| 2 | processor | Huawei | no recommendation | High | 9000.0 |
| 3 | camera | Lenovo | no recommendation | Medium | 3000.0 |
| 4 | battery | Samsung | By company | Low | 1500.0 |

Model Number:

Accessory Name:

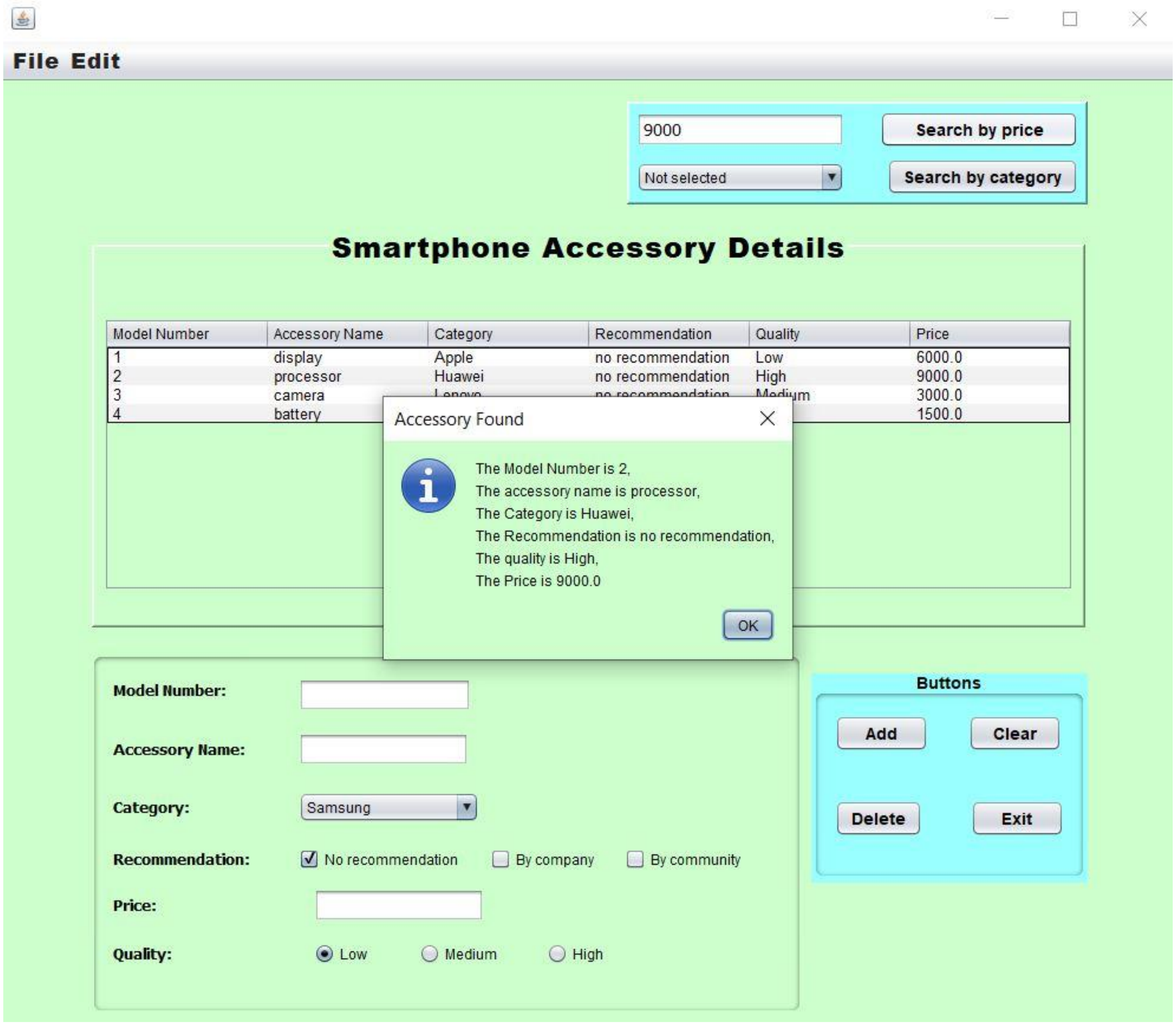
Category: Samsung

Recommendation: ☑ No recommendation □ By company □ By community

Price:

Quality: ● Low ○ Medium ○ High

*Figure 8: Click on search button*



9000

Not selected

**Smartphone Accessory Details**

| Model Number | Accessory Name | Category | Recommendation | Quality | Price |
|---|---|---|---|---|---|
| 1 | display | Apple | no recommendation | Low | 6000.0 |
| 2 | processor | Huawei | no recommendation | High | 9000.0 |
| 3 | camera | Lenovo | no recommendation | Medium | 3000.0 |
| 4 | battery | | | | 1500.0 |

Accessory Found

The Model Number is 2,
The accessory name is processor,
The Category is Huawei,
The Recommendation is no recommendation,
The quality is High,
The Price is 9000.0

Model Number:

Accessory Name:

Category: Samsung

Recommendation: ☑ No recommendation ☐ By company ☐ By community

Price:

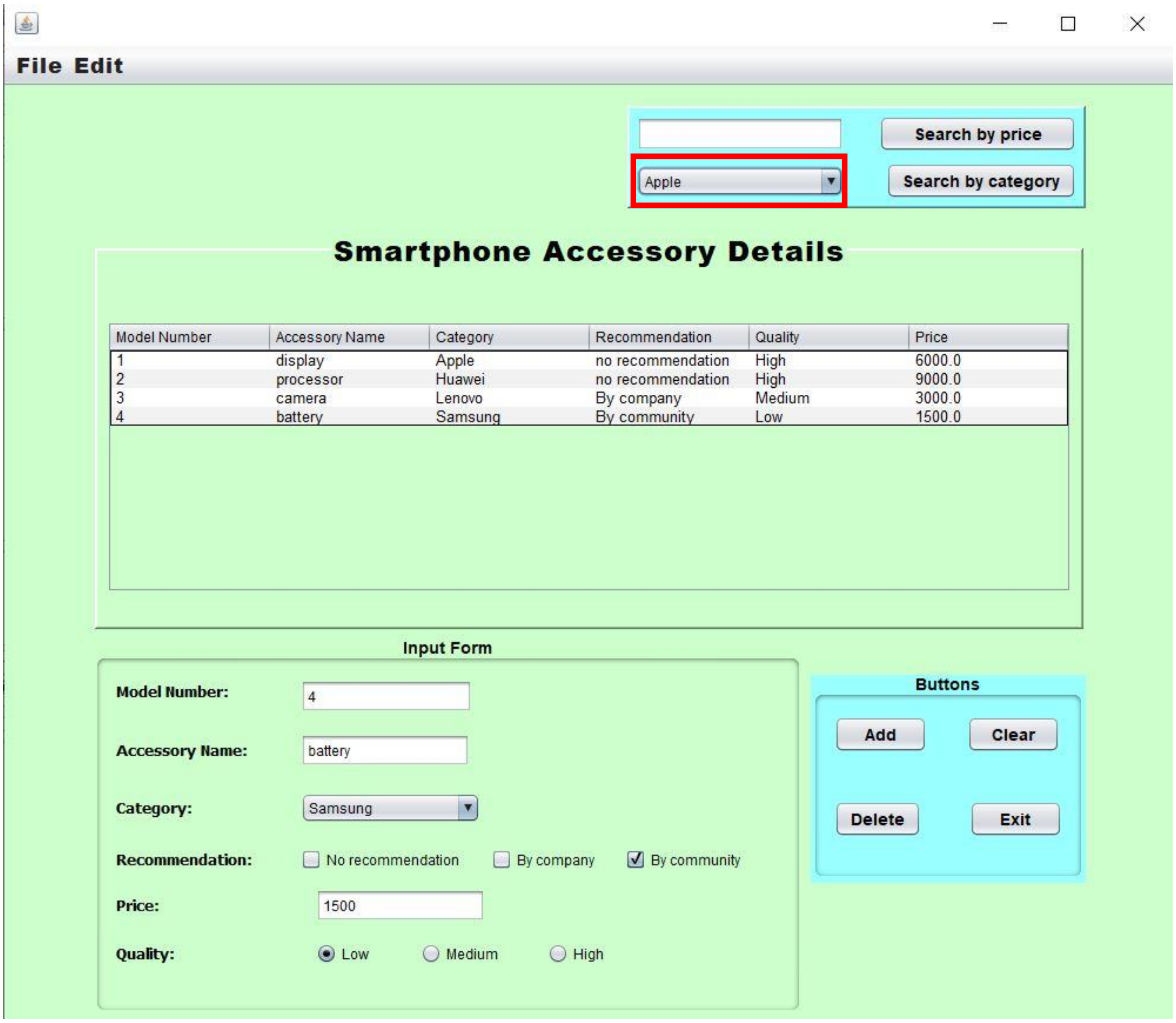
Quality: ● Low ○ Medium ○ High

*Figure 9: Dislpay the details of select priced item*

### 5.2.3 Test B (3): searching for items based on category

| | |
|---|---|
| **Objectives** | Search item based on category |
| **Action** | Select category<br>Category = “Apple”<br>Click on search button |
| **Expected Result** | Accessories Details will be shown in Dialog Box |
| **Actual Result** | Accessories Details is shown in Dialog Box |
| **Conclusion** | Test was successful |

*Table 4: Test B (3): searching for items based on category*



| Model Number | Accessory Name | Category | Recommendation | Quality | Price |
|---|---|---|---|---|---|
| 1 | display | Apple | no recommendation | High | 6000.0 |
| 2 | processor | Huawei | no recommendation | High | 9000.0 |
| 3 | camera | Lenovo | By company | Medium | 3000.0 |
| 4 | battery | Samsung | By community | Low | 1500.0 |

Model Number: 4

Accessory Name: battery

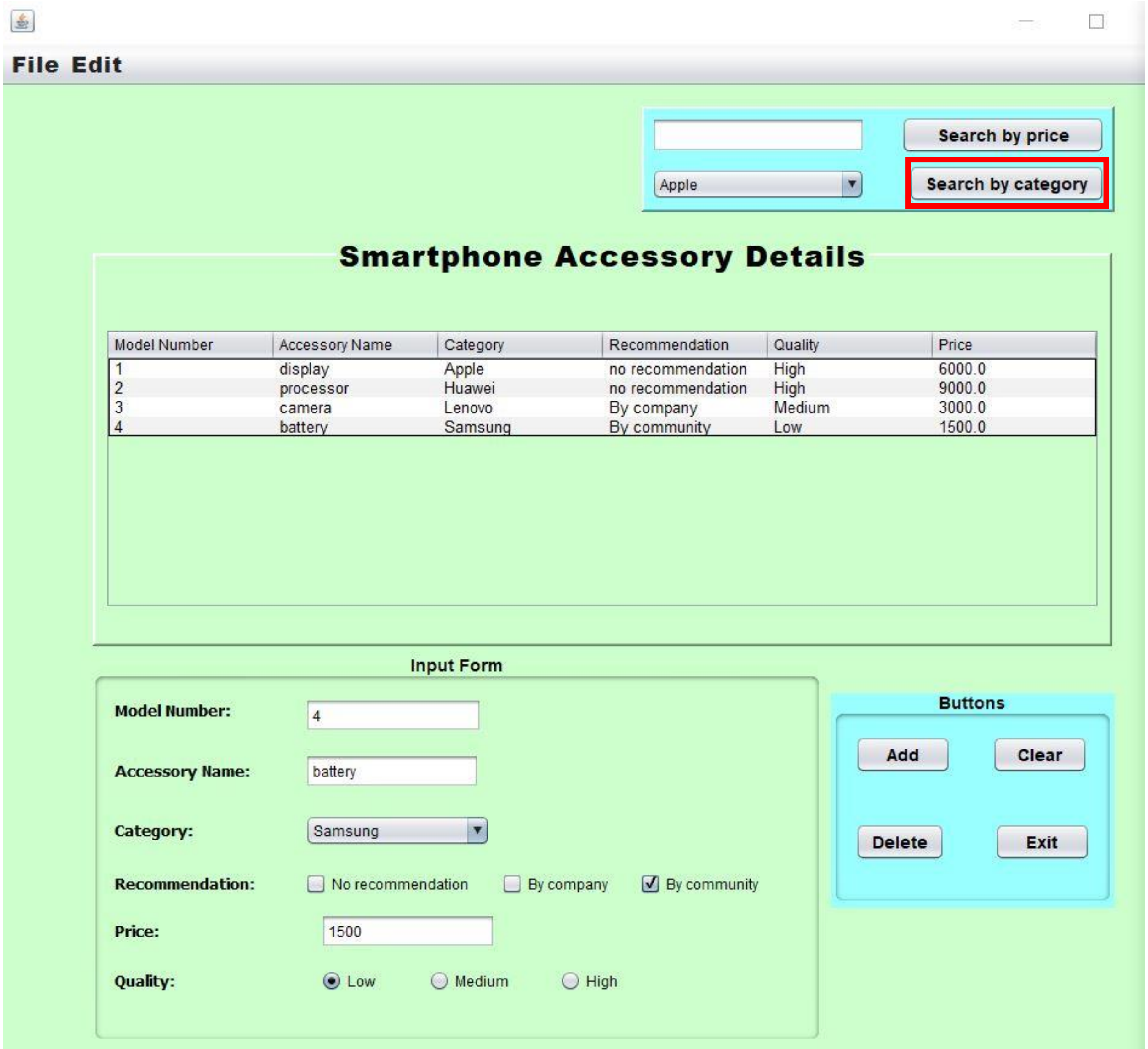
Category: Samsung

Recommendation: □ No recommendation □ By company ☑ By community

Price: 1500

Quality: ● Low ○ Medium ○ High

*Figure 10: select the category*



Apple

Smartphone Accessory Details

| Model Number | Accessory Name | Category | Recommendation | Quality | Price |
|---|---|---|---|---|---|
| 1 | display | Apple | no recommendation | High | 6000.0 |
| 2 | processor | Huawei | no recommendation | High | 9000.0 |
| 3 | camera | Lenovo | By company | Medium | 3000.0 |
| 4 | battery | Samsung | By community | Low | 1500.0 |

Input Form

Model Number: 4

Accessory Name: battery

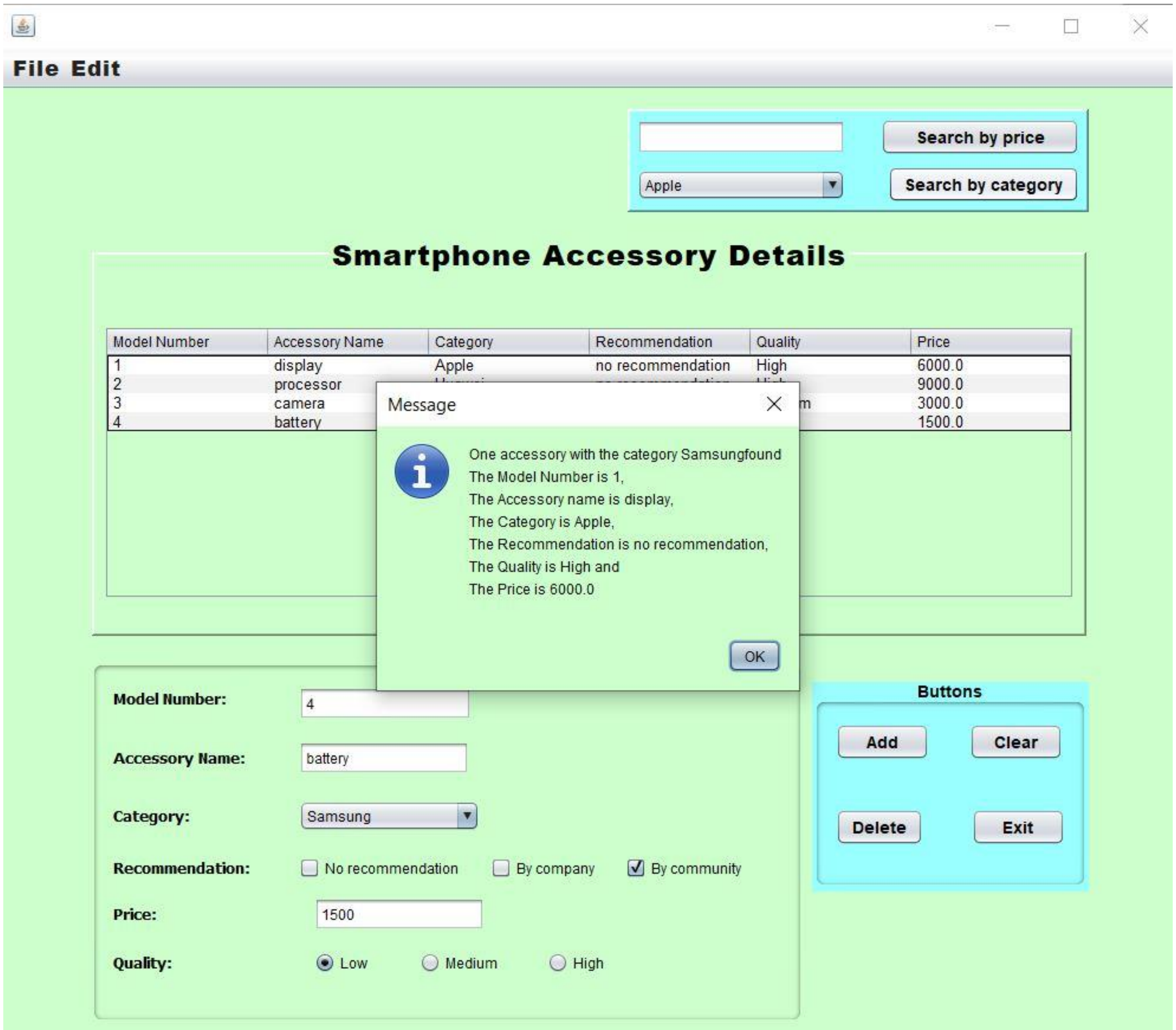
Category: Samsung

Recommendation: ☐ No recommendation ☐ By company ☑ By community

Price: 1500

Quality: ● Low ○ Medium ○ High

*Figure 11: click on search button*



Apple

Smartphone Accessory Details

| Model Number | Accessory Name | Category | Recommendation | Quality | Price |
|---|---|---|---|---|---|
| 1 | display | Apple | no recommendation | High | 6000.0 |
| 2 | processor | | | | 9000.0 |
| 3 | camera | | | m | 3000.0 |
| 4 | battery | | | | 1500.0 |

Message

One accessory with the category Samsungfound
The Model Number is 1,
The Accessory name is display,
The Category is Apple,
The Recommendation is no recommendation,
The Quality is High and
The Price is 6000.0

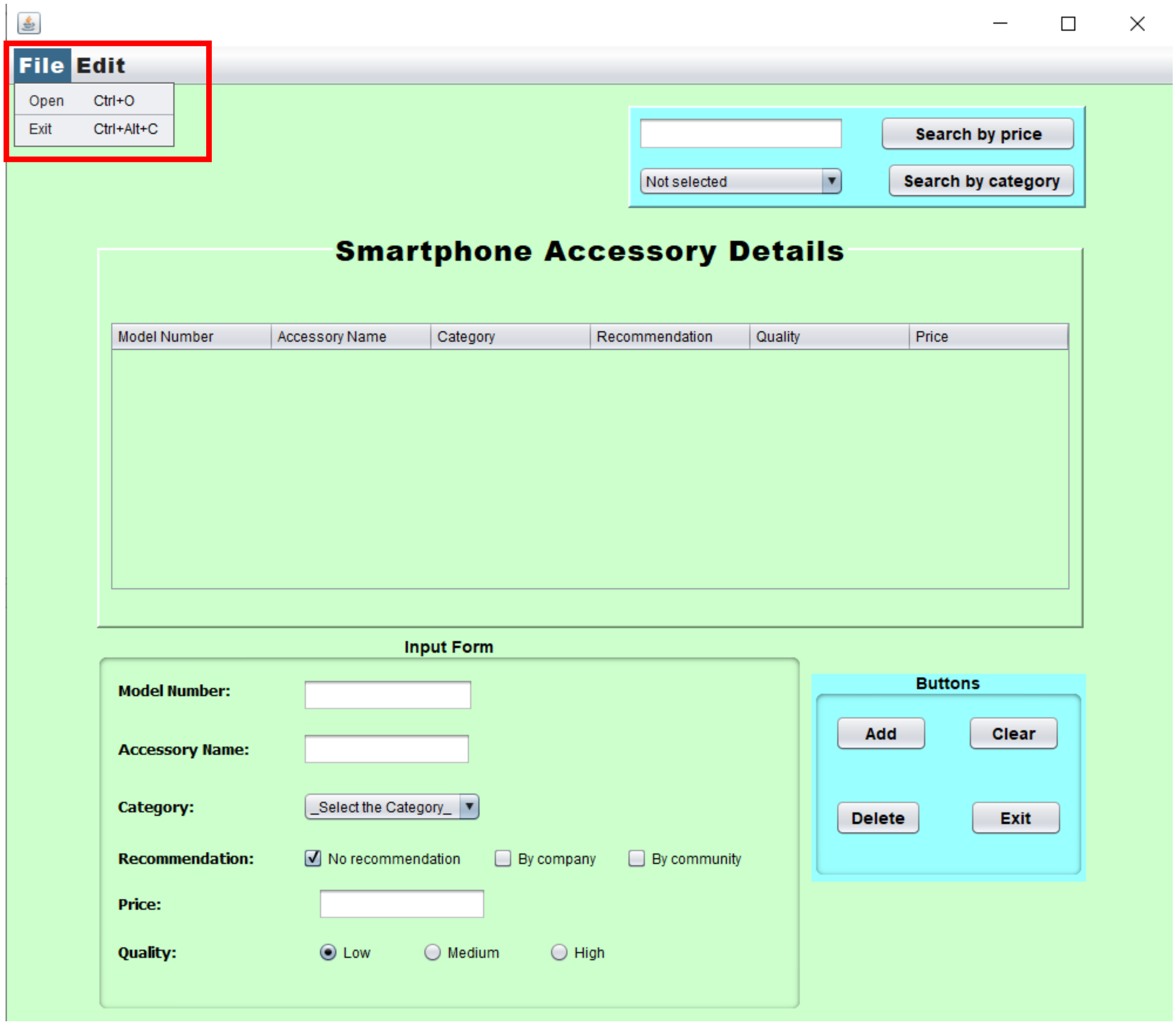
| | |
|---|---|
| Model Number: | 4 |
| Accessory Name: | battery |
| Category: | Samsung |
| Recommendation: | □ No recommendation □ By company ☑ By community |
| Price: | 1500 |
| Quality: | ● Low ○ Medium ○ High |

*Figure 12: display the details of selected category*

### 5.2.4 Test B (4): Opening a file from menu

| | |
|---|---|
| **Objectives** | Opening a file from menu |
| **Action** | Select File menu<br>Select open<br>Display file in excel<br>Go to help menu<br>Select help<br>Display read me file |
| **Expected Result** | File will open |
| **Actual Result** | File is open |
| **Conclusion** | Test was successful |

*Table 5: Test B (4): Opening a file from menu*



Not selected ▼

| Model Number | Accessory Name | Category | Recommendation | Quality | Price |
|---|---|---|---|---|---|

Model Number:

Accessory Name:

Category: _Select the Category_ ▼

Recommendation: ☑ No recommendation ☐ By company ☐ By community

Price:

Quality: ● Low ○ Medium ○ High

*Figure 13: select open from menu bar*

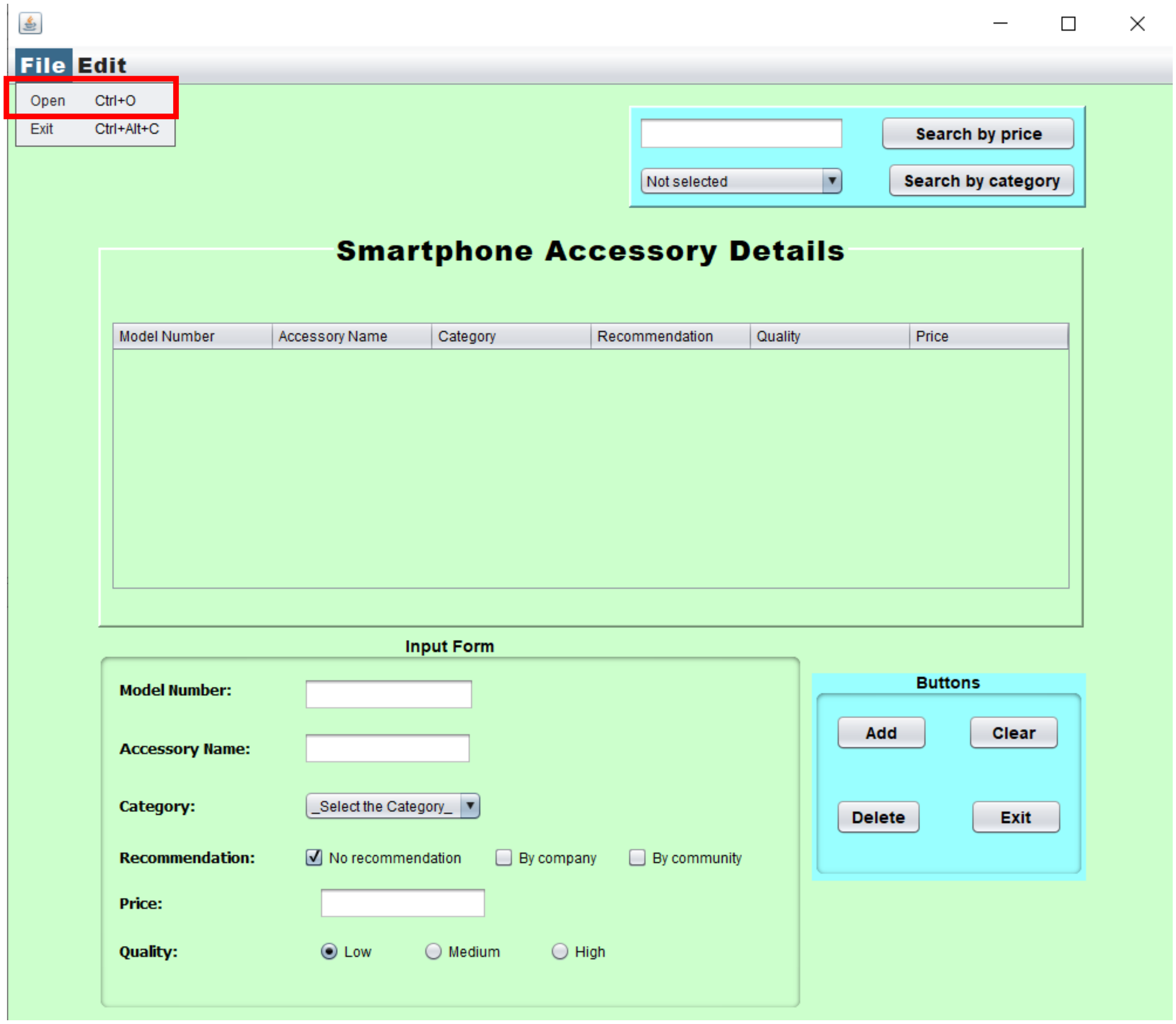
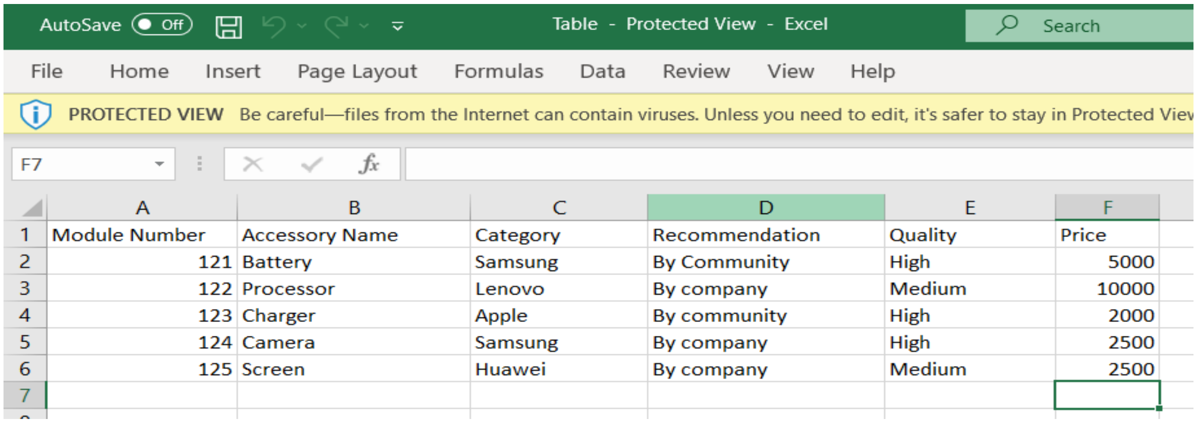


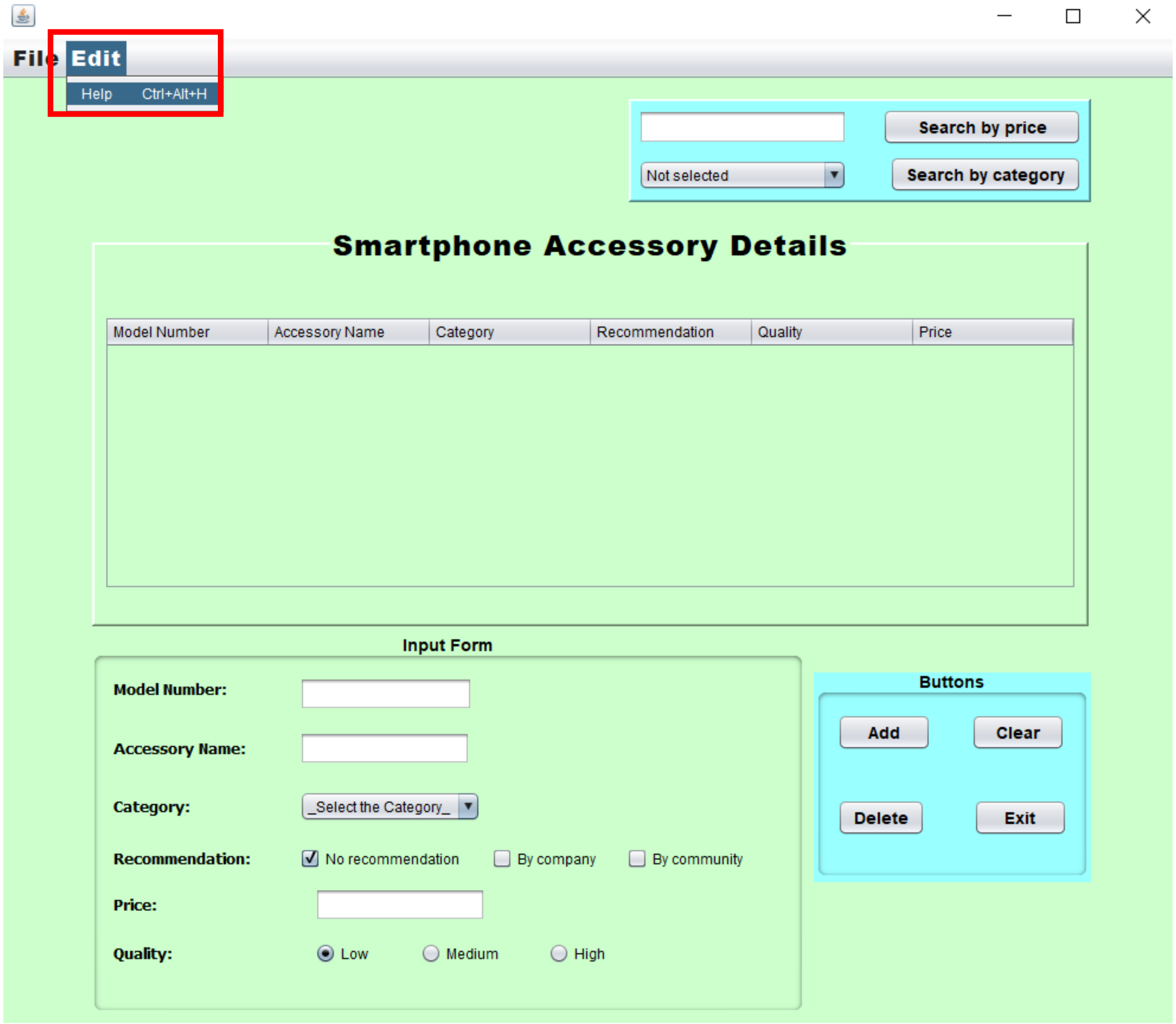
*Figure 14:select Open*



PROTECTED VIEW Be careful—files from the Internet can contain viruses. Unless you need to edit, it's safer to stay in Protected View

| | A | B | C | D | E | F |
|---|---|---|---|---|---|---|
| 1 | Module Number | Accessory Name | Category | Recommendation | Quality | Price |
| 2 | 121 | Battery | Samsung | By Community | High | 5000 |
| 3 | 122 | Processor | Lenovo | By company | Medium | 10000 |
| 4 | 123 | Charger | Apple | By community | High | 2000 |
| 5 | 124 | Camera | Samsung | By company | High | 2500 |
| 6 | 125 | Screen | Huawei | By company | Medium | 2500 |
| 7 | | | | | | |

*Figure 15: file display*



Not selected

| Model Number | Accessory Name | Category | Recommendation | Quality | Price |
|---|---|---|---|---|---|

Model Number:

Accessory Name:

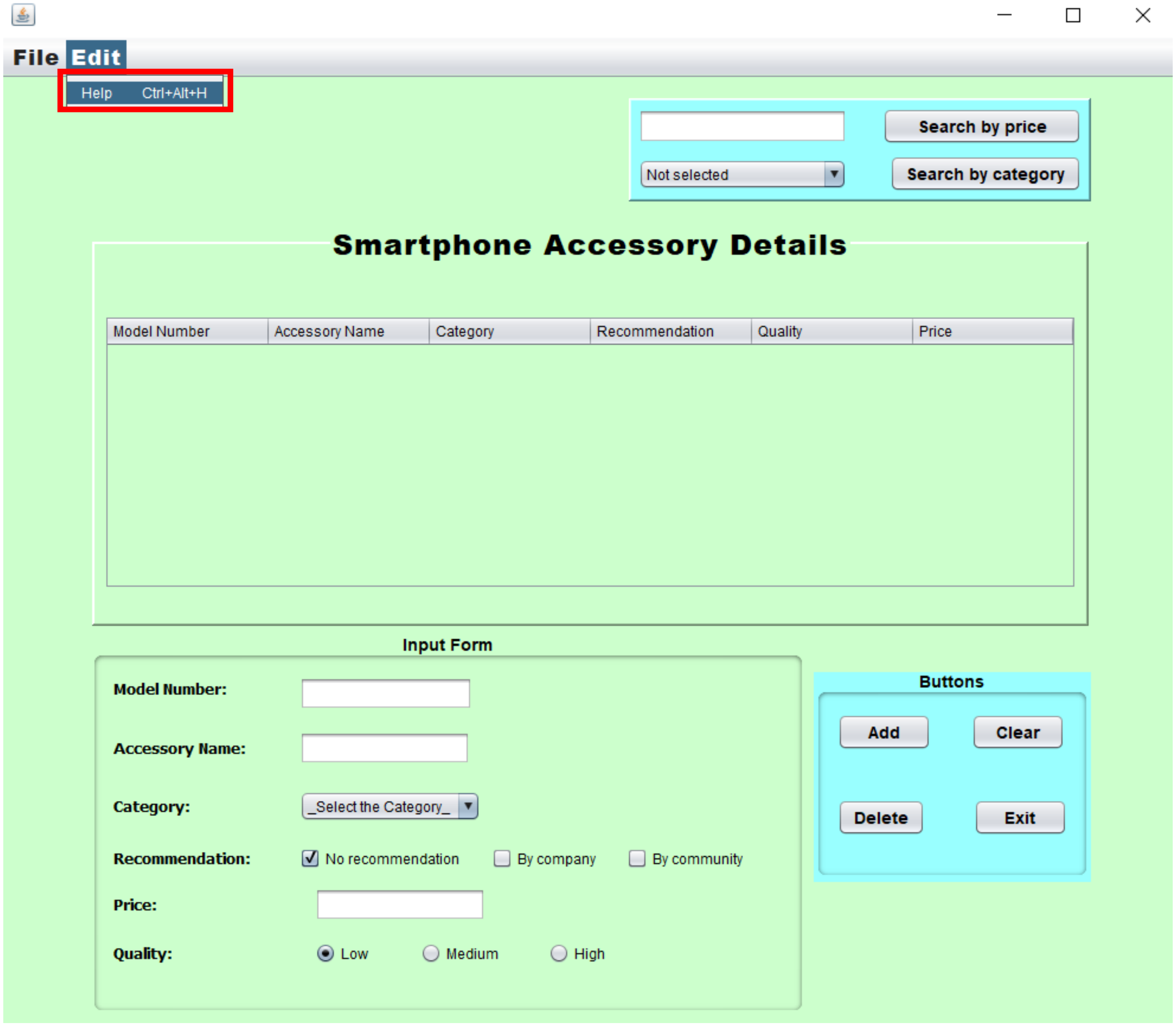
Category: _Select the Category_

Recommendation: ☑ No recommendation □ By company □ By community

Price:

Quality: ● Low ○ Medium ○ High

*Figure 16: go to edit*



Not selected

Model Number:

Accessory Name:

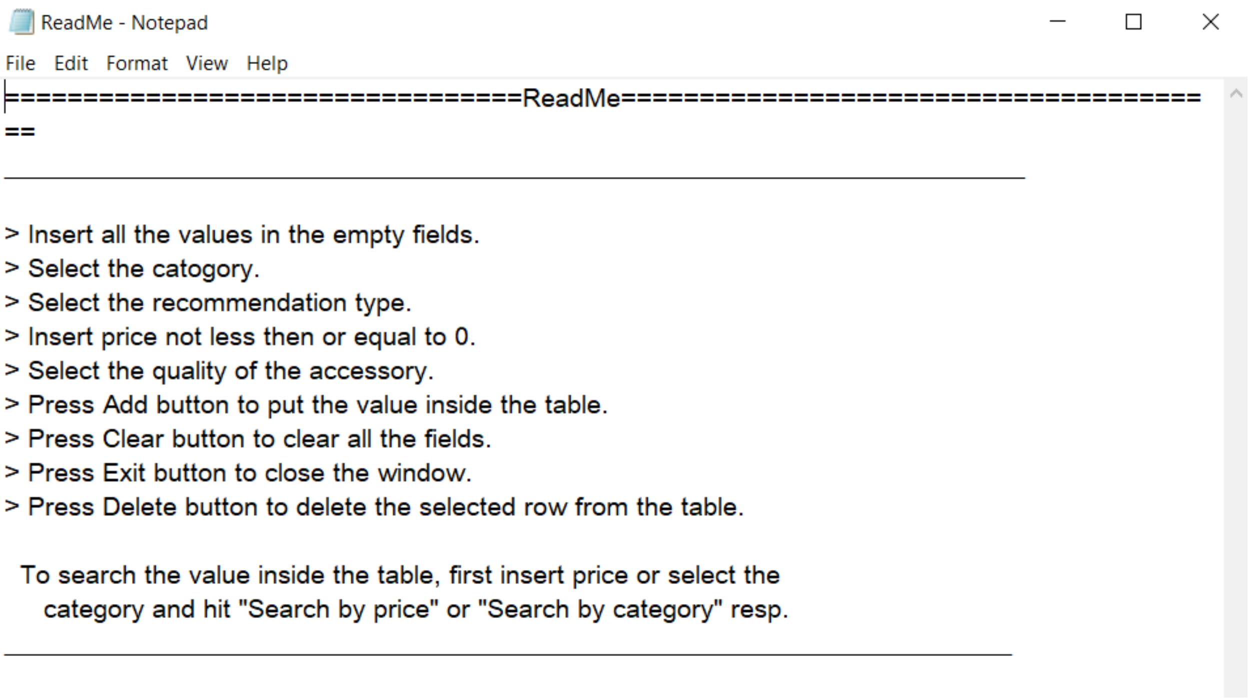
Category: _Select the Category_

Recommendation: ☑ No recommendation ☐ By company ☐ By community

Price:

Quality: ● Low ○ Medium ○ High

*Figure 17: Select help*



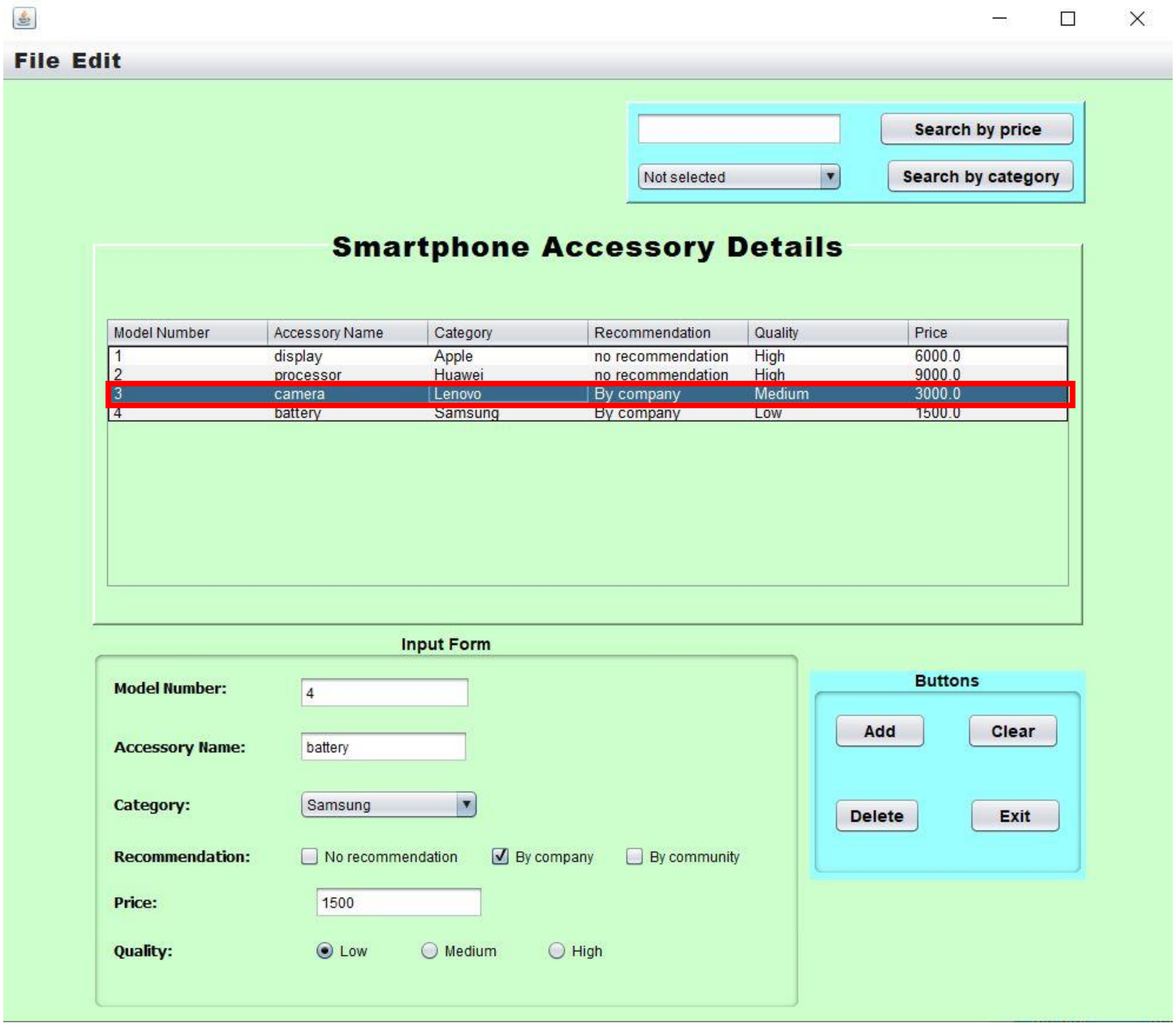
```
================================ReadMe=====================================
==
________________________________________________________________________

> Insert all the values in the empty fields.
> Select the catogory.
> Select the recommendation type.
> Insert price not less then or equal to 0.
> Select the quality of the accessory.
> Press Add button to put the value inside the table.
> Press Clear button to clear all the fields.
> Press Exit button to close the window.
> Press Delete button to delete the selected row from the table.

  To search the value inside the table, first insert price or select the
    category and hit "Search by price" or "Search by category" resp.
________________________________________________________________________
```

*Figure 18: Display Read me file*

### 5.2.5 Test B (5): Deleting an item from table

| | |
|---|---|
| **Objectives** | Delete item from table |
| **Action** | Select a row on table<br>Click the delete button |
| **Expected Result** | Item will be deleted |
| **Actual Result** | Item is deleted |
| **Conclusion** | Test was successful |

*Table 6: Test B (5): Deleting an item from table*



Smartphone Accessory Details

| Model Number | Accessory Name | Category | Recommendation | Quality | Price |
|---|---|---|---|---|---|
| 1 | display | Apple | no recommendation | High | 6000.0 |
| 2 | processor | Huawei | no recommendation | High | 9000.0 |
| 3 | camera | Lenovo | By company | Medium | 3000.0 |
| 4 | battery | Samsung | By company | Low | 1500.0 |

Model Number: 4

Accessory Name: battery

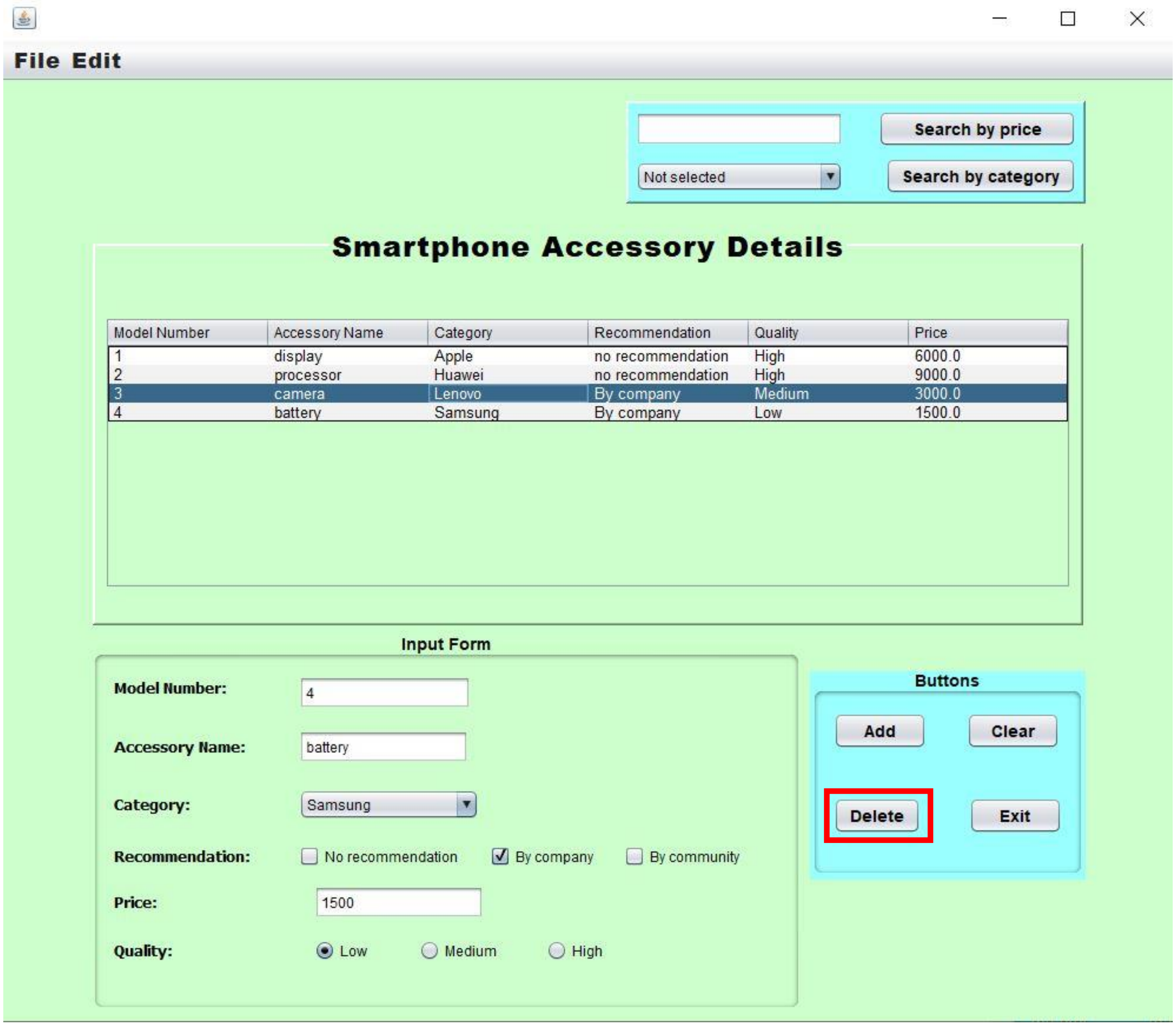
Category: Samsung

Recommendation: □ No recommendation ☑ By company □ By community

Price: 1500

Quality: ● Low ○ Medium ○ High

*Figure 19: select a row in a table*



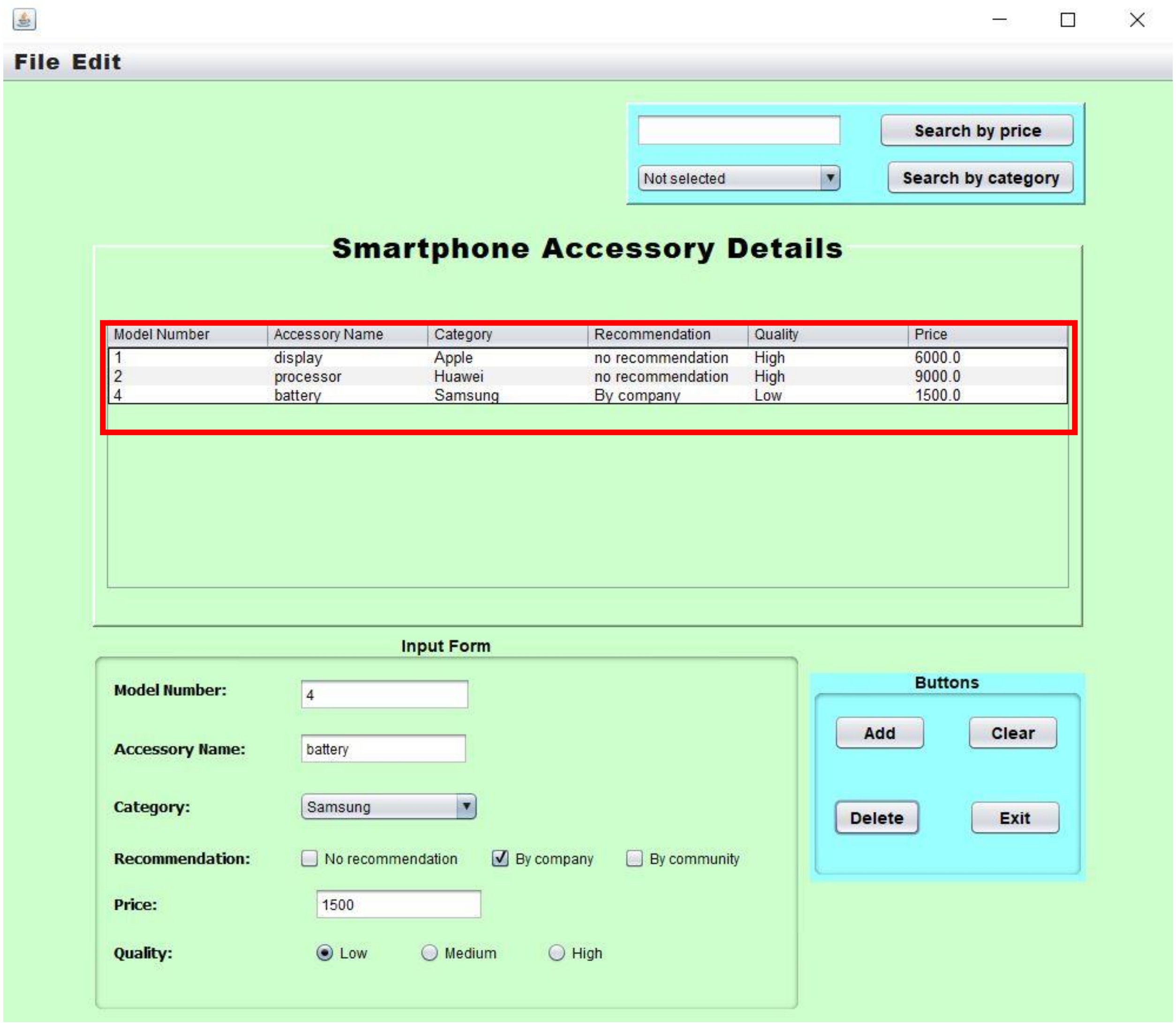
Smartphone Accessory Details

| Model Number | Accessory Name | Category | Recommendation | Quality | Price |
|---|---|---|---|---|---|
| 1 | display | Apple | no recommendation | High | 6000.0 |
| 2 | processor | Huawei | no recommendation | High | 9000.0 |
| 3 | camera | Lenovo | By company | Medium | 3000.0 |
| 4 | battery | Samsung | By company | Low | 1500.0 |

Input Form

| | |
|---|---|
| Model Number: | 4 |
| Accessory Name: | battery |
| Category: | Samsung |
| Recommendation: | □ No recommendation ☑ By company □ By community |
| Price: | 1500 |
| Quality: | ● Low ○ Medium ○ High |

*Figure 20: Click on delete button*



Not selected

Smartphone Accessory Details

| Model Number | Accessory Name | Category | Recommendation | Quality | Price |
|---|---|---|---|---|---|
| 1 | display | Apple | no recommendation | High | 6000.0 |
| 2 | processor | Huawei | no recommendation | High | 9000.0 |
| 4 | battery | Samsung | By company | Low | 1500.0 |

Input Form

Model Number: 4

Accessory Name: battery

Category: Samsung

Recommendation: □ No recommendation ☑ By company □ By community

Price: 1500

Quality: ● Low ○ Medium ○ High

Buttons

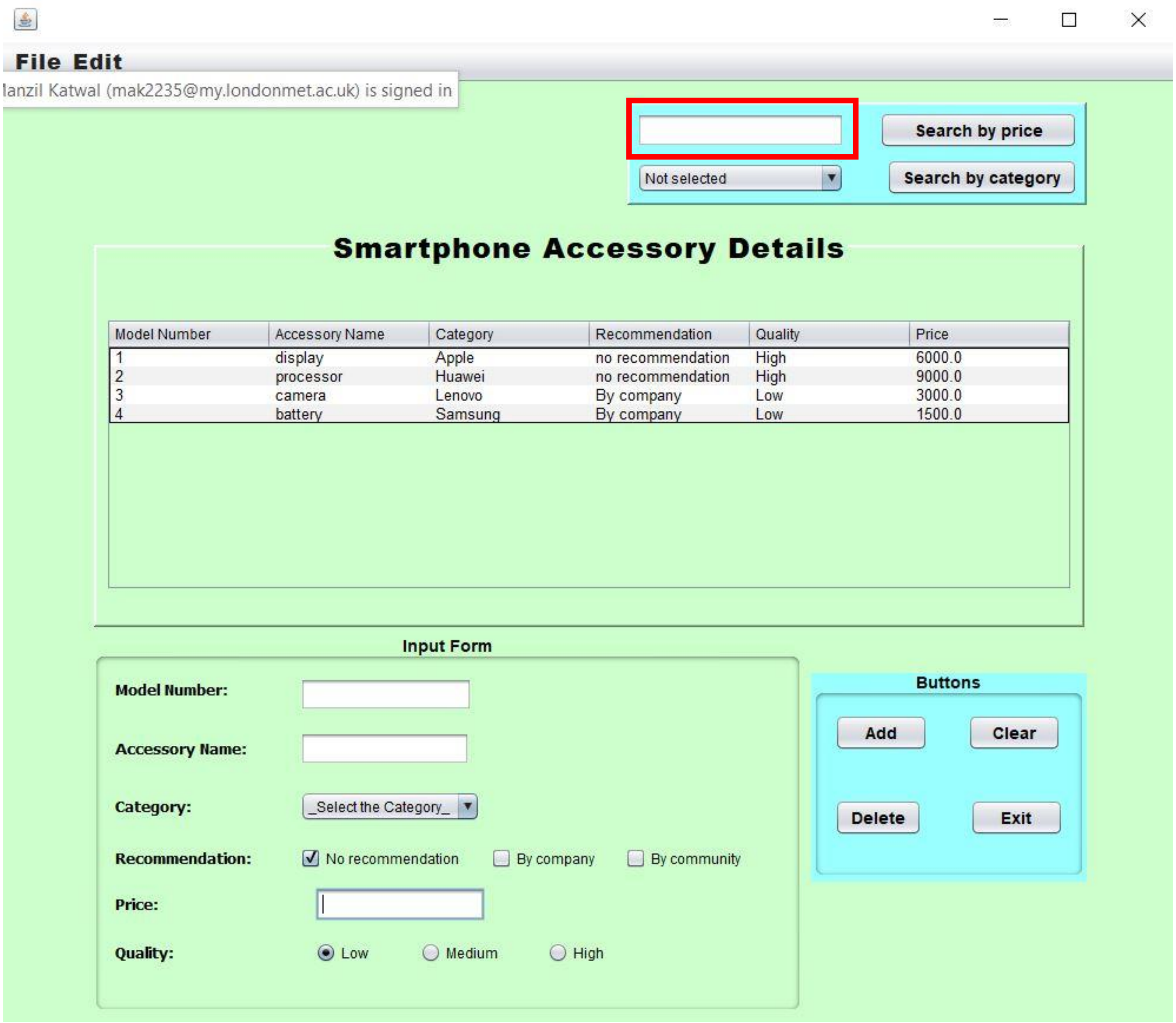
*Figure 21: selected row is deleted*

## 5.3 Testing Scenario C**: Evidence on system validation program**

### 5.3.1 Test C (1): Shows Dialog Box for not inserting the value in Input text field

| | |
|---|---|
| **Objectives** | Display Dialog Box for not inserting the value in Input text field |
| **Action** | empty value on Input text field<br>Model Number = “”<br>Accessories Name = “”<br>Category = “”<br>Recommendation =””<br>Price =<br>Quality =<br>Click add button<br>Display dialog box “Please enter details in all fields” |
| **Expected Result** | Dialog Box will Display |
| **Actual Result** | Dialog Box is Display |
| **Conclusion** | Test was successful |

*Table 7: Test C (1): Shows Dialog Box for not inserting the value in Input text field*



lanzil Katwal (mak2235@my.londonmet.ac.uk) is signed in

Not selected

Smartphone Accessory Details

| Model Number | Accessory Name | Category | Recommendation | Quality | Price |
|---|---|---|---|---|---|
| 1 | display | Apple | no recommendation | High | 6000.0 |
| 2 | processor | Huawei | no recommendation | High | 9000.0 |
| 3 | camera | Lenovo | By company | Low | 3000.0 |
| 4 | battery | Samsung | By company | Low | 1500.0 |

Input Form

Model Number:

Accessory Name:

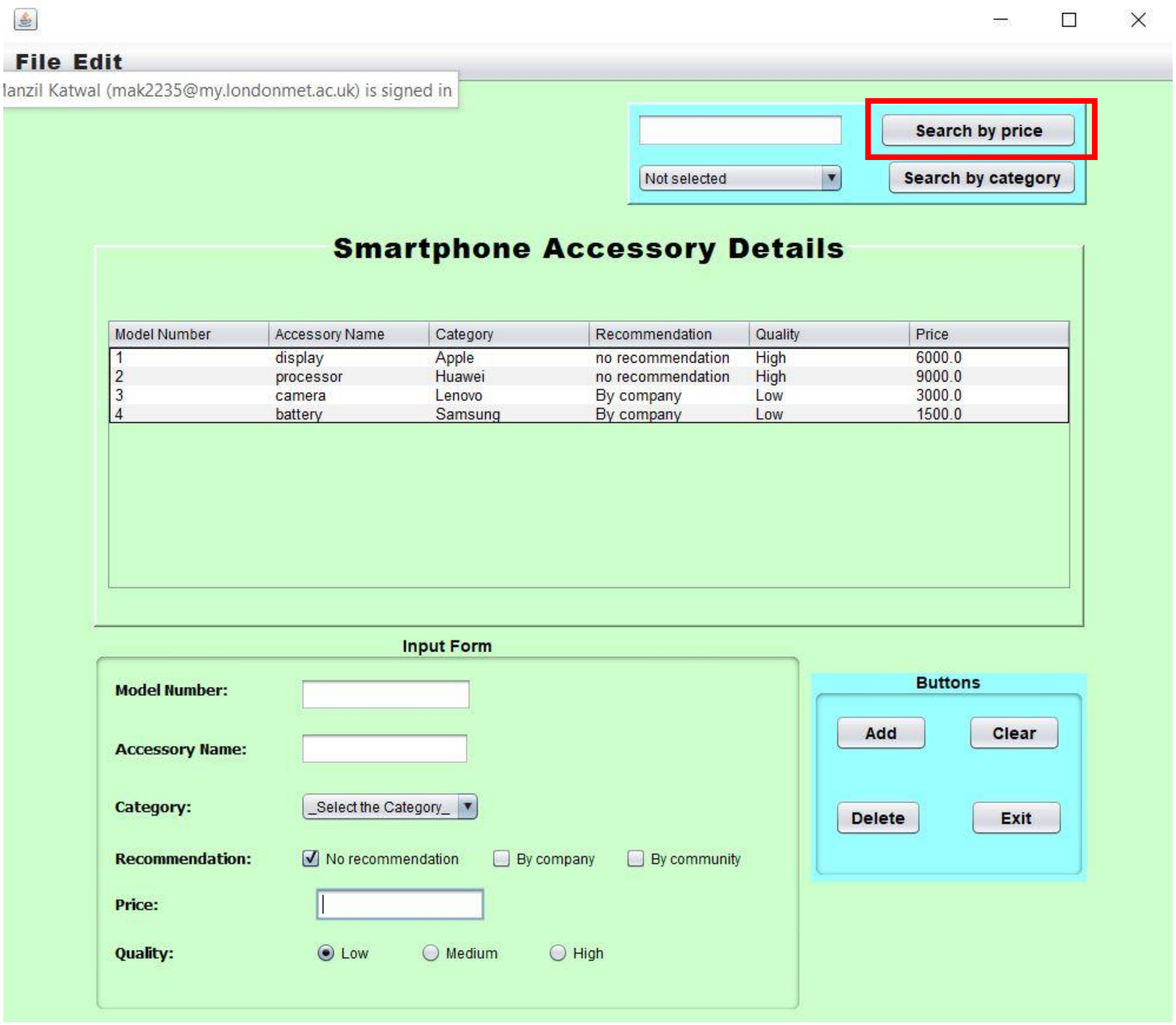
Category: _Select the Category_

Recommendation: ☑ No recommendation □ By company □ By community

Price:

Quality: ● Low ○ Medium ○ High

*Figure 22: empty text fields*



| Model Number | Accessory Name | Category | Recommendation | Quality | Price |
|---|---|---|---|---|---|
| 1 | display | Apple | no recommendation | High | 6000.0 |
| 2 | processor | Huawei | no recommendation | High | 9000.0 |
| 3 | camera | Lenovo | By company | Low | 3000.0 |
| 4 | battery | Samsung | By company | Low | 1500.0 |

Model Number:

Accessory Name:

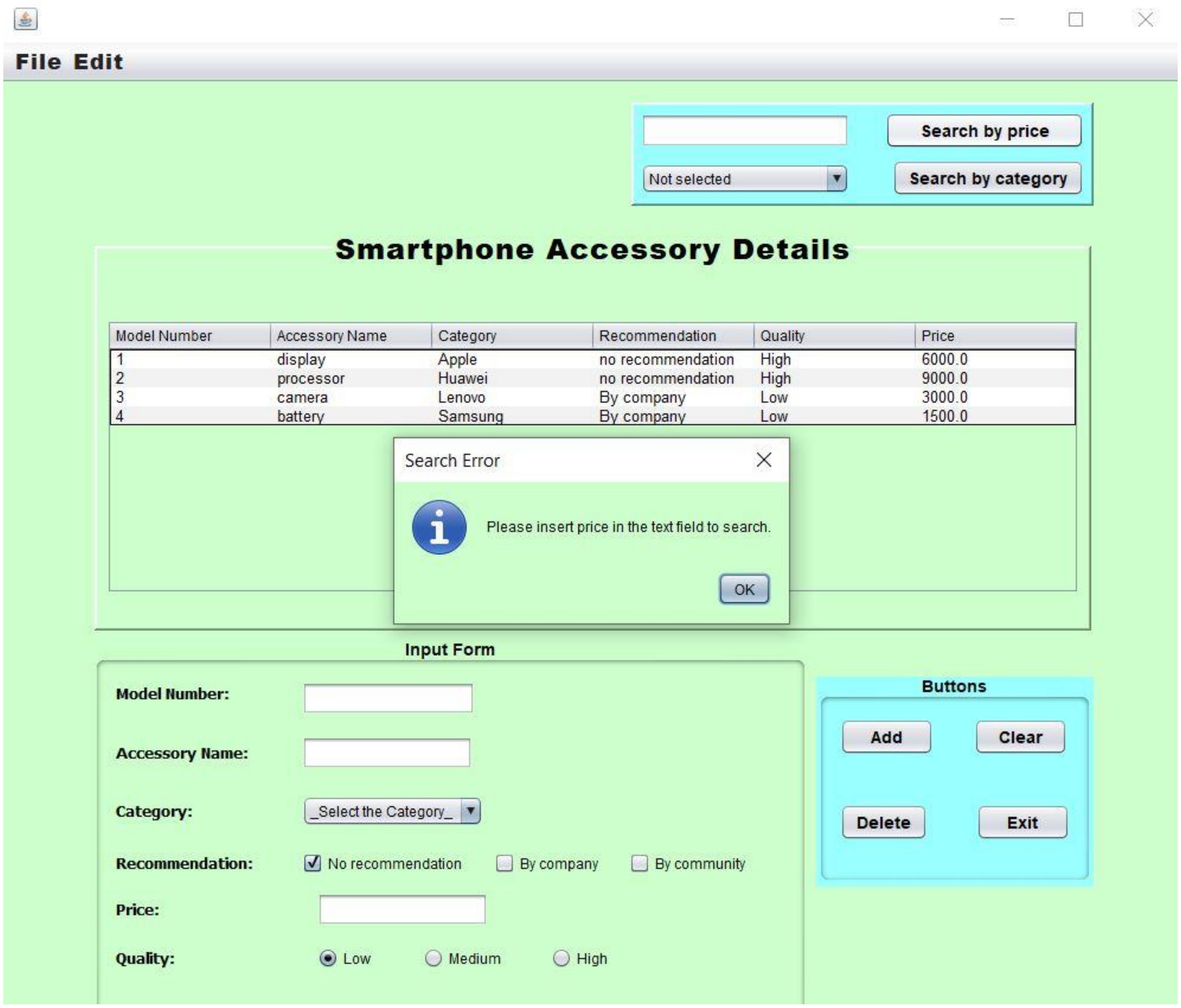
Category: _Select the Category_

Recommendation: ☑ No recommendation □ By company □ By community

Price:

Quality: ● Low ○ Medium ○ High

*Figure 23: Click search button*



Not selected

| Model Number | Accessory Name | Category | Recommendation | Quality | Price |
|---|---|---|---|---|---|
| 1 | display | Apple | no recommendation | High | 6000.0 |
| 2 | processor | Huawei | no recommendation | High | 9000.0 |
| 3 | camera | Lenovo | By company | Low | 3000.0 |
| 4 | battery | Samsung | By company | Low | 1500.0 |

Search Error

Please insert price in the text field to search.

Model Number:

Accessory Name:

Category: _Select the Category_

Recommendation: ☑ No recommendation ☐ By company ☐ By community

Price:

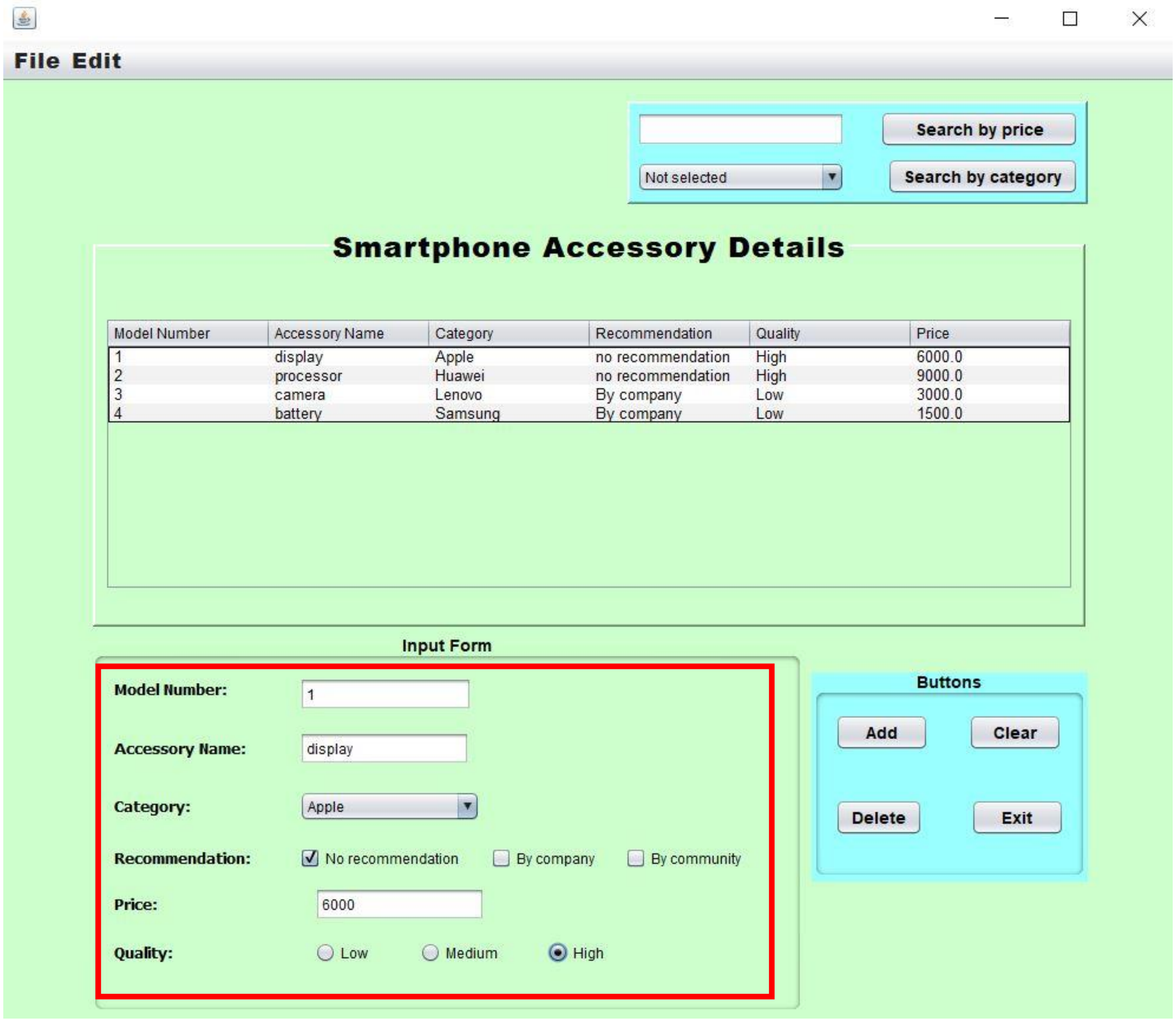
Quality: ● Low ○ Medium ○ High

*Figure 24: Display Error message*

### 5.3.2 Test C (2): Shows Dialog Box for inserting the already used model number in text field

| | |
|---|---|
| **Objectives** | Display Dialog Box inserting the already used model number value in text field. |
| **Action** | Model number = 1<br>Accessory name = “Display”<br>Category = “Apple”<br>Recommendation = no recommendation<br>Price = 6000<br>Quality = high<br>Repeat the same model number next time<br>Display the error message “the model number is already used in the table” |
| **Expected Result** | Dialog Box will Display |
| **Actual Result** | Dialog Box is Displayed |
| **Conclusion** | Test was successful |

*Table 8:Test C (2): Shows Dialog Box for inserting the already used model number in text field*



Smartphone Accessory Details

| Model Number | Accessory Name | Category | Recommendation | Quality | Price |
|---|---|---|---|---|---|
| 1 | display | Apple | no recommendation | High | 6000.0 |
| 2 | processor | Huawei | no recommendation | High | 9000.0 |
| 3 | camera | Lenovo | By company | Low | 3000.0 |
| 4 | battery | Samsung | By company | Low | 1500.0 |

Model Number: 1

Accessory Name: display

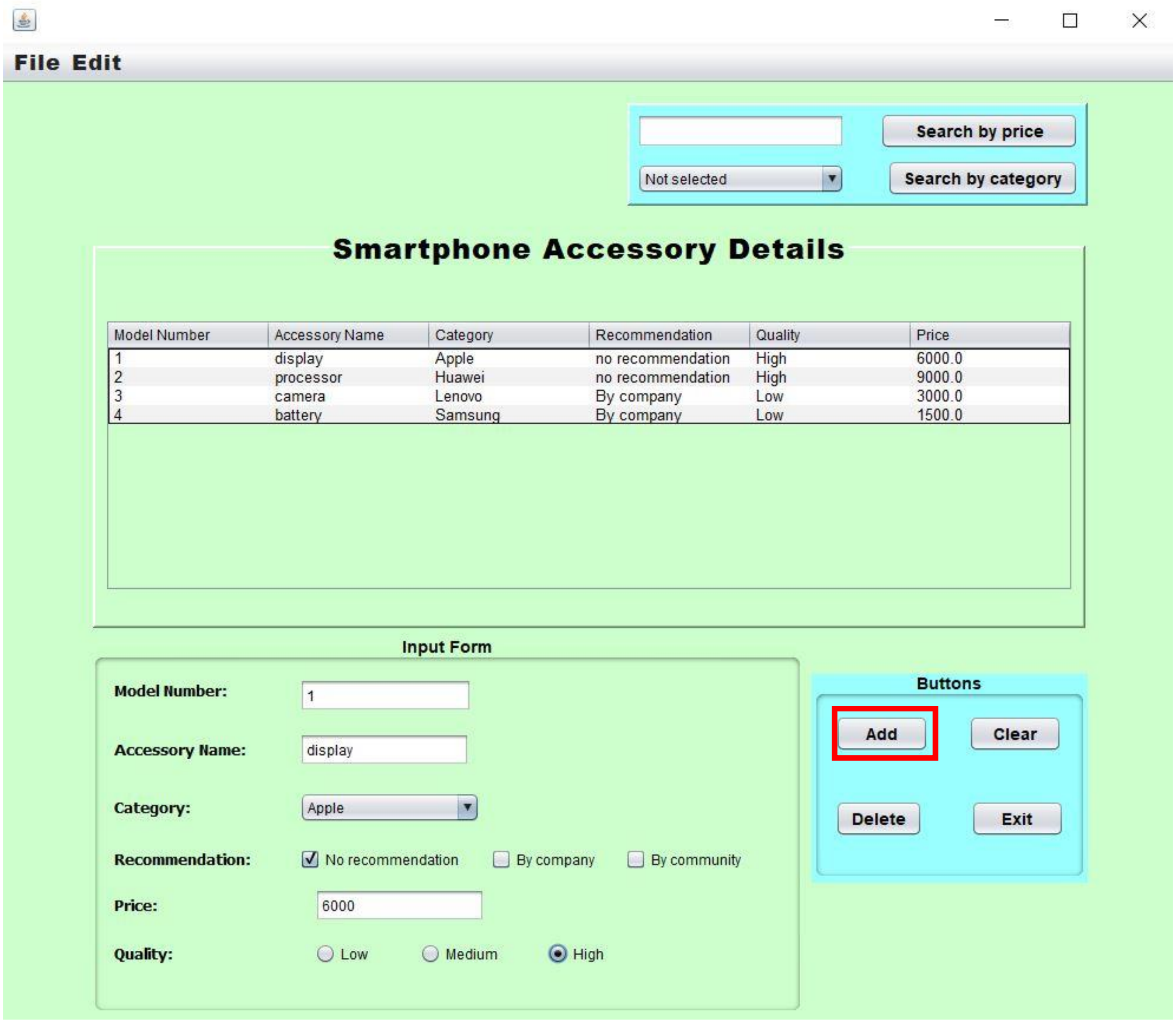
Category: Apple

Recommendation: ☑ No recommendation □ By company □ By community

Price: 6000

Quality: ○ Low ○ Medium ● High

*Figure 25: insert the same values in the text fields*



Smartphone Accessory Details

| Model Number | Accessory Name | Category | Recommendation | Quality | Price |
|---|---|---|---|---|---|
| 1 | display | Apple | no recommendation | High | 6000.0 |
| 2 | processor | Huawei | no recommendation | High | 9000.0 |
| 3 | camera | Lenovo | By company | Low | 3000.0 |
| 4 | battery | Samsung | By company | Low | 1500.0 |

Input Form

Model Number: 1

Accessory Name: display

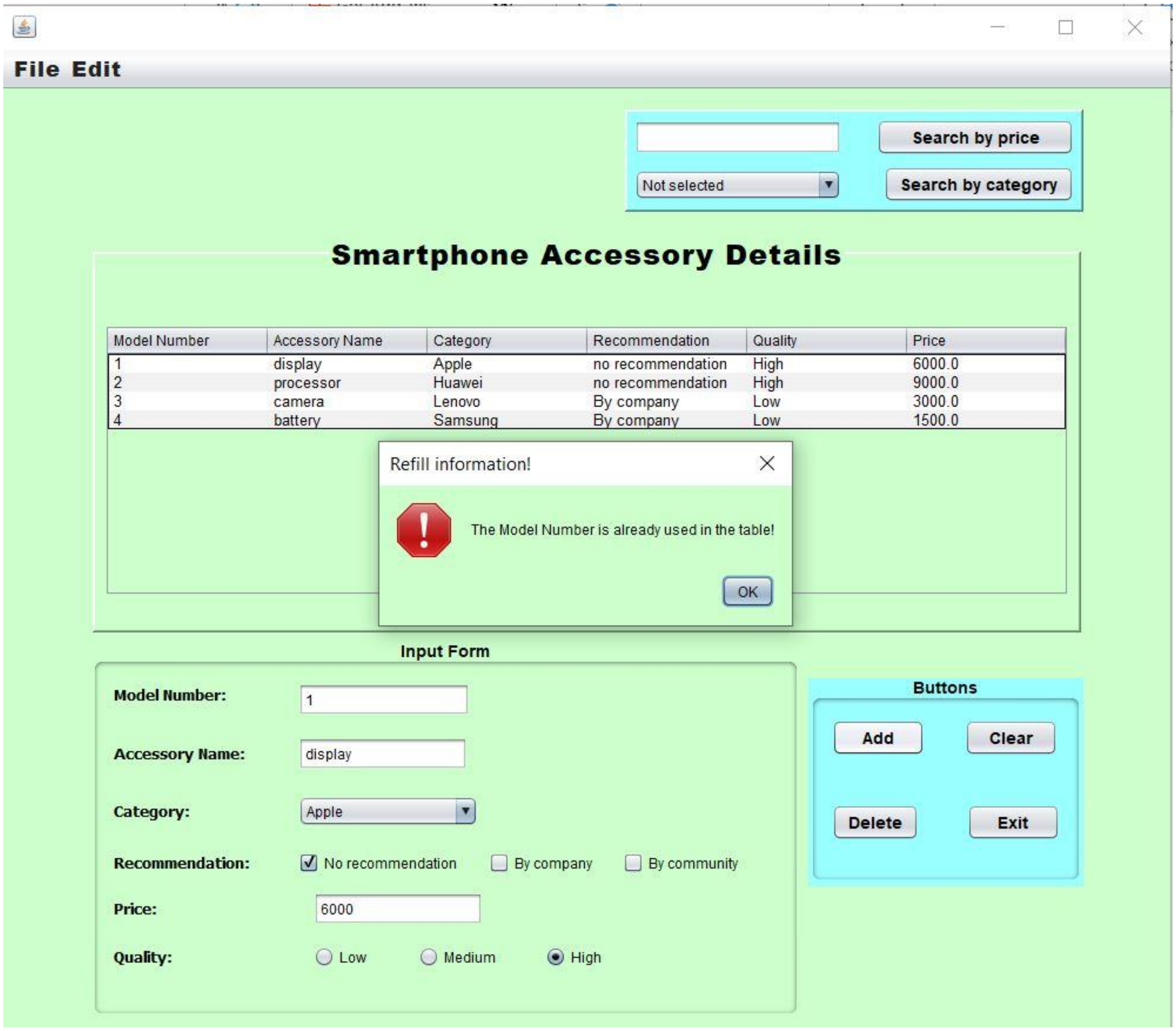
Category: Apple

Recommendation: ☑ No recommendation ☐ By company ☐ By community

Price: 6000

Quality: ○ Low ○ Medium ● High

*Figure 26: Click Add button*



| Model Number | Accessory Name | Category | Recommendation | Quality | Price |
|---|---|---|---|---|---|
| 1 | display | Apple | no recommendation | High | 6000.0 |
| 2 | processor | Huawei | no recommendation | High | 9000.0 |
| 3 | camera | Lenovo | By company | Low | 3000.0 |
| 4 | battery | Samsung | By company | Low | 1500.0 |

Refill information!

The Model Number is already used in the table!

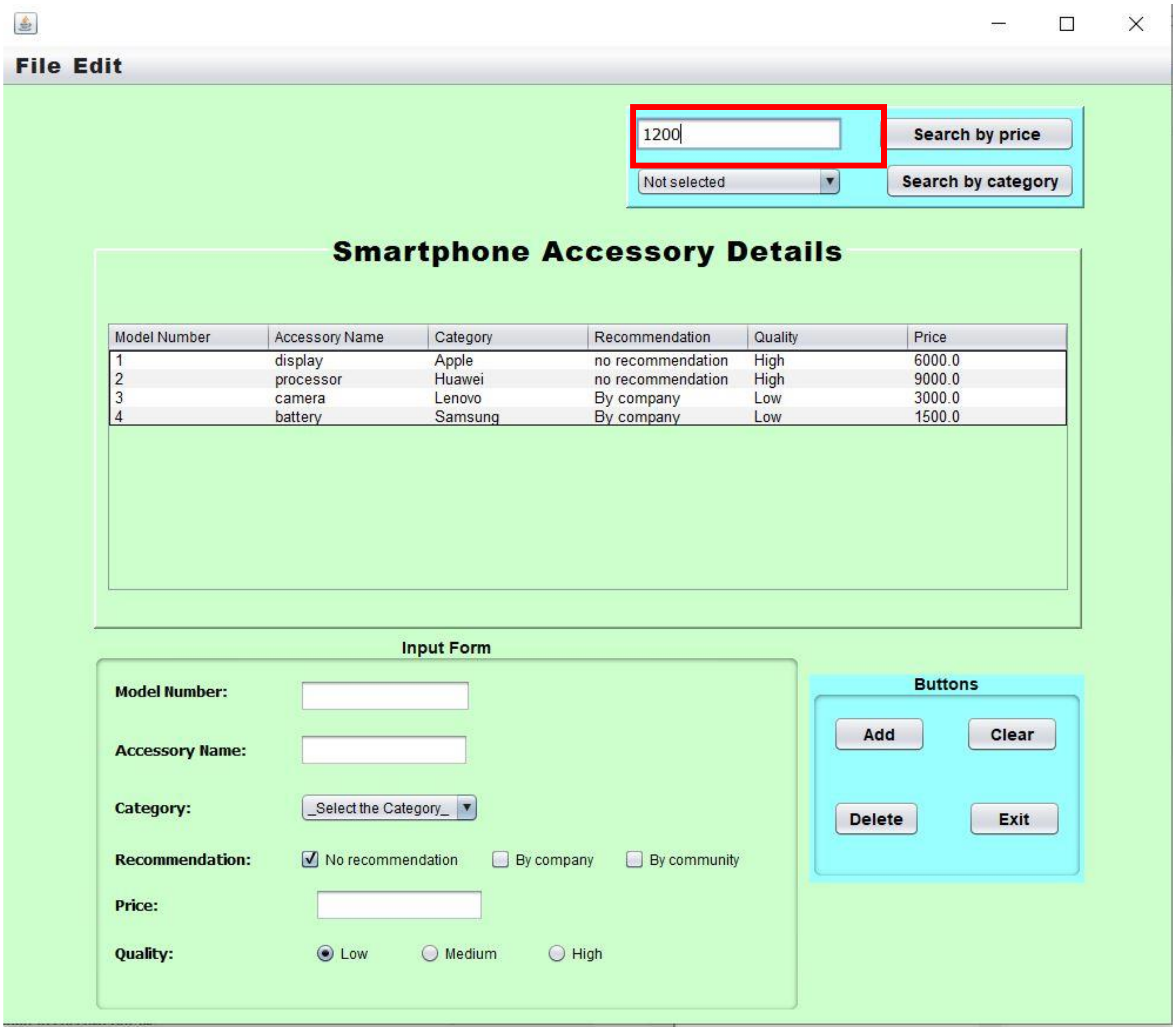
| | |
|---|---|
| Model Number: | 1 |
| Accessory Name: | display |
| Category: | Apple |
| Recommendation: | ☑ No recommendation ☐ By company ☐ By community |
| Price: | 6000 |
| Quality: | ○ Low ○ Medium ● High |

*Figure 27: Displays the dialog box of error message*

### 5.3.3 Test C (3): Shows Dialog Box for inserting the wrong value in the text field.

| | |
|---|---|
| **Objectives** | Display Dialog Box inserting the already used model number value in text field. |
| **Action** | empty value on Search text field<br>Price = 1200<br>Click Search Button<br>Display Dialog Box “Accessory of price 1200 is not found” |
| **Expected Result** | Dialog Box will Display |
| **Actual Result** | Dialog Box is Displayed |
| **Conclusion** | Test was successful |

*Table 9: Test C (3): Shows Dialog Box for inserting the wrong value in the text field.*



1200

Not selected

**Smartphone Accessory Details**

| Model Number | Accessory Name | Category | Recommendation | Quality | Price |
|---|---|---|---|---|---|
| 1 | display | Apple | no recommendation | High | 6000.0 |
| 2 | processor | Huawei | no recommendation | High | 9000.0 |
| 3 | camera | Lenovo | By company | Low | 3000.0 |
| 4 | battery | Samsung | By company | Low | 1500.0 |

Input Form

Model Number:

Accessory Name:

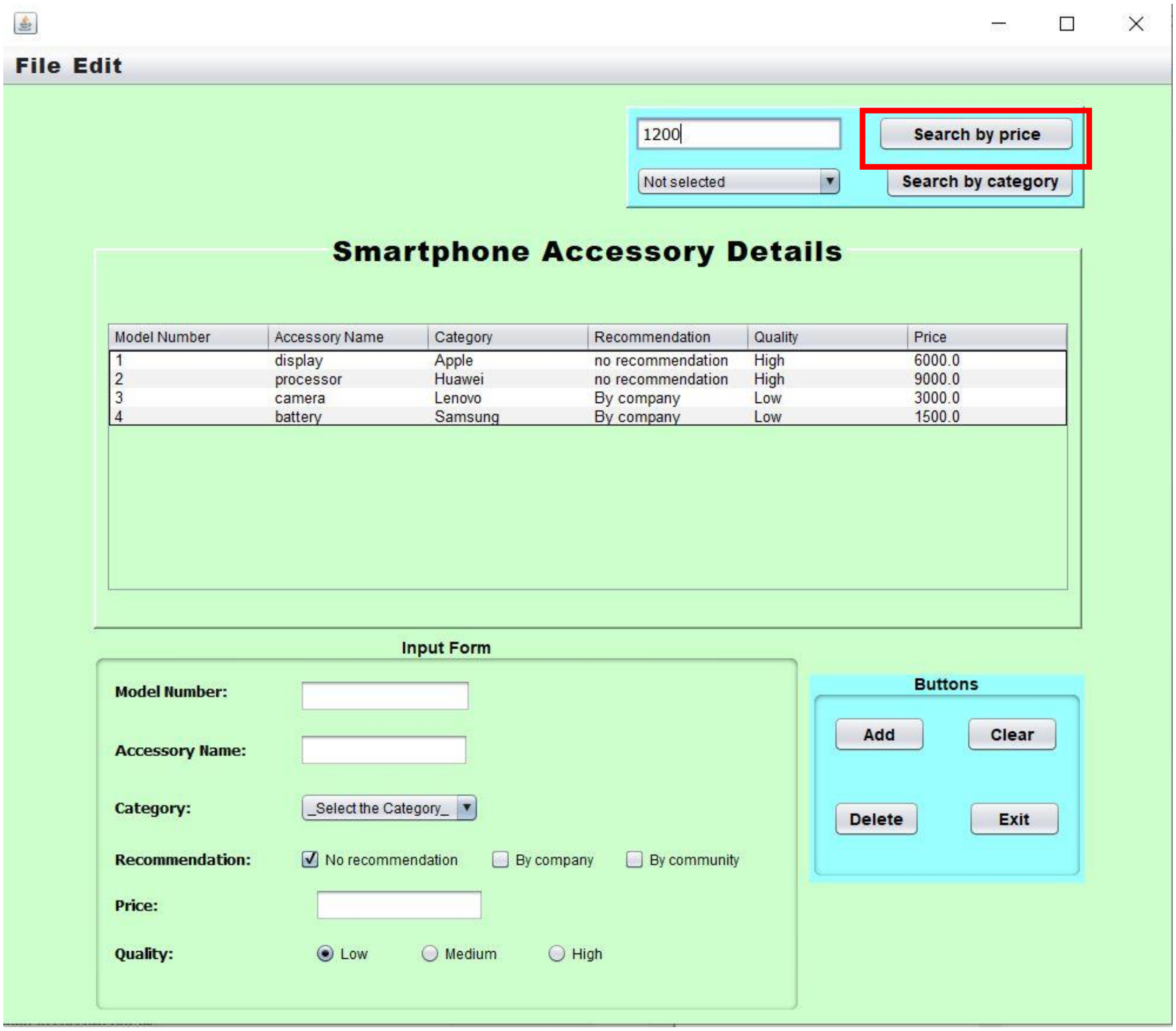
Category: _Select the Category_

Recommendation: ☑ No recommendation ☐ By company ☐ By community

Price:

Quality: ● Low ○ Medium ○ High

*Figure 28: insert the non-existent price*



1200

Not selected

| Model Number | Accessory Name | Category | Recommendation | Quality | Price |
|---|---|---|---|---|---|
| 1 | display | Apple | no recommendation | High | 6000.0 |
| 2 | processor | Huawei | no recommendation | High | 9000.0 |
| 3 | camera | Lenovo | By company | Low | 3000.0 |
| 4 | battery | Samsung | By company | Low | 1500.0 |

Model Number:

Accessory Name:

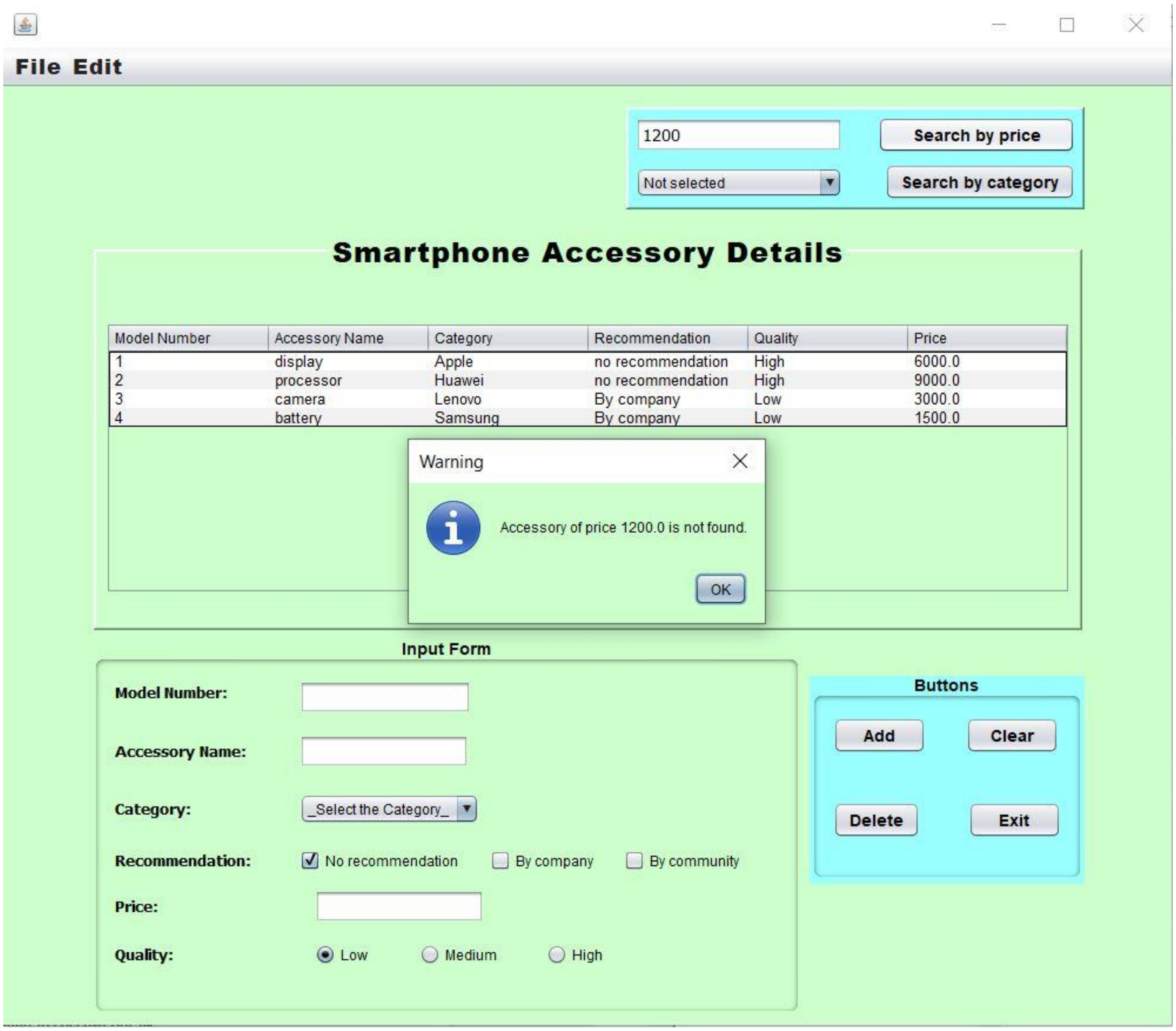
Category: _Select the Category_

Recommendation: ☑ No recommendation ☐ By company ☐ By community

Price:

Quality: ● Low ○ Medium ○ High

*Figure 29: click on search button*



1200

Not selected

| Model Number | Accessory Name | Category | Recommendation | Quality | Price |
|---|---|---|---|---|---|
| 1 | display | Apple | no recommendation | High | 6000.0 |
| 2 | processor | Huawei | no recommendation | High | 9000.0 |
| 3 | camera | Lenovo | By company | Low | 3000.0 |
| 4 | battery | Samsung | By company | Low | 1500.0 |

Accessory of price 1200.0 is not found.

Model Number:

Accessory Name:

Category: _Select the Category_

Recommendation: ☑ No recommendation ☐ By company ☐ By community

Price:

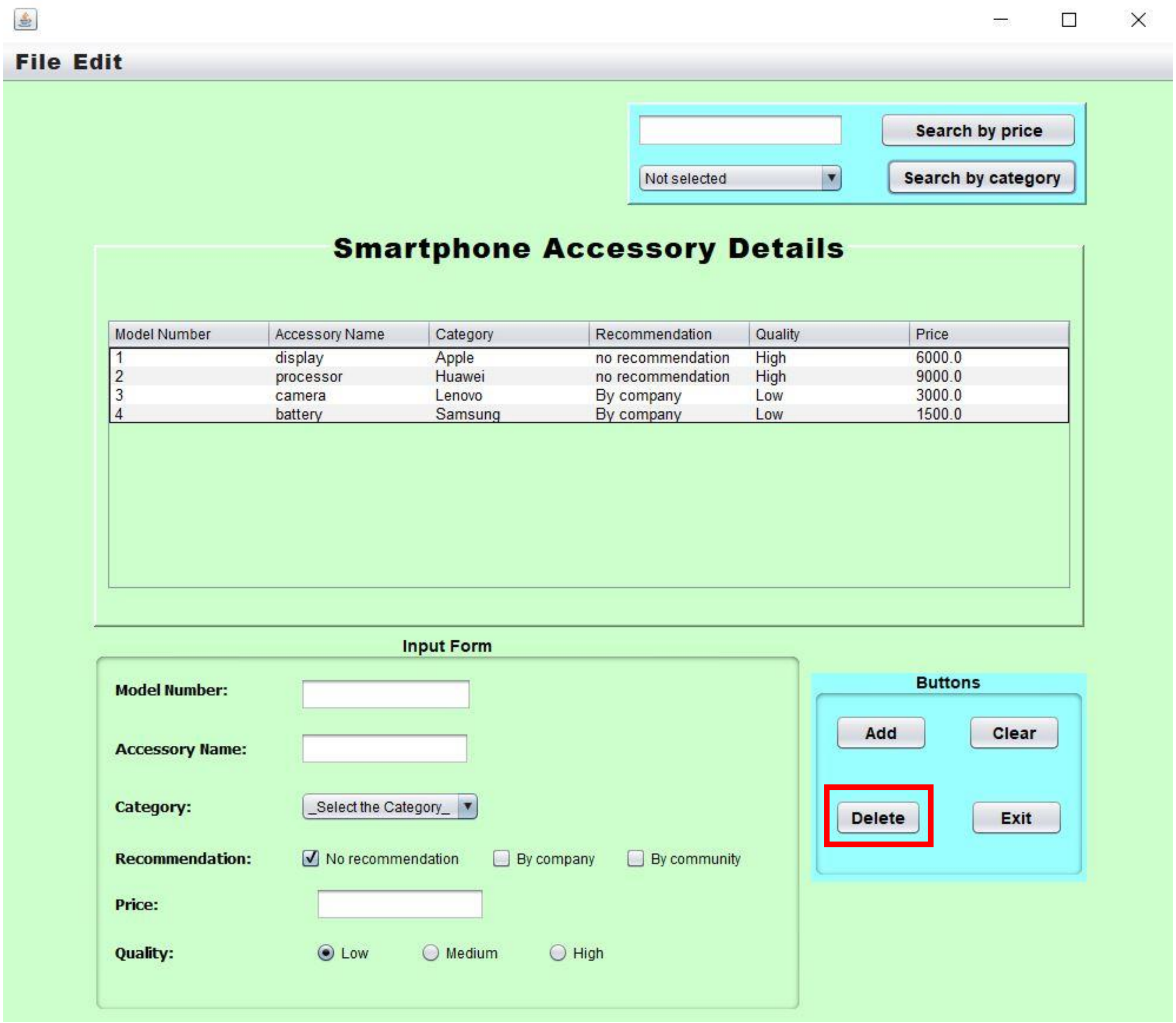
Quality: ● Low ○ Medium ○ High

*Figure 30: displays warning message*

### 5.3.4 Test C (4): Shows Dialog Box for not selecting a row while clicking delete button

| | |
|---|---|
| **Objectives** | Display Dialog Box for not selecting a row while clicking delete button |
| **Action** | Click Delete Button<br>Display Dialog Box "Please Select a Row You want to Delete!" |
| **Expected Result** | Dialog Box will Display |
| **Actual Result** | Dialog Box is Display |
| **Conclusion** | Test was successful |

*Table 10: Test C (4): Shows Dialog Box for not selecting a row while clicking delete button*



Smartphone Accessory Details

| Model Number | Accessory Name | Category | Recommendation | Quality | Price |
|---|---|---|---|---|---|
| 1 | display | Apple | no recommendation | High | 6000.0 |
| 2 | processor | Huawei | no recommendation | High | 9000.0 |
| 3 | camera | Lenovo | By company | Low | 3000.0 |
| 4 | battery | Samsung | By company | Low | 1500.0 |

Input Form

Model Number:

Accessory Name:

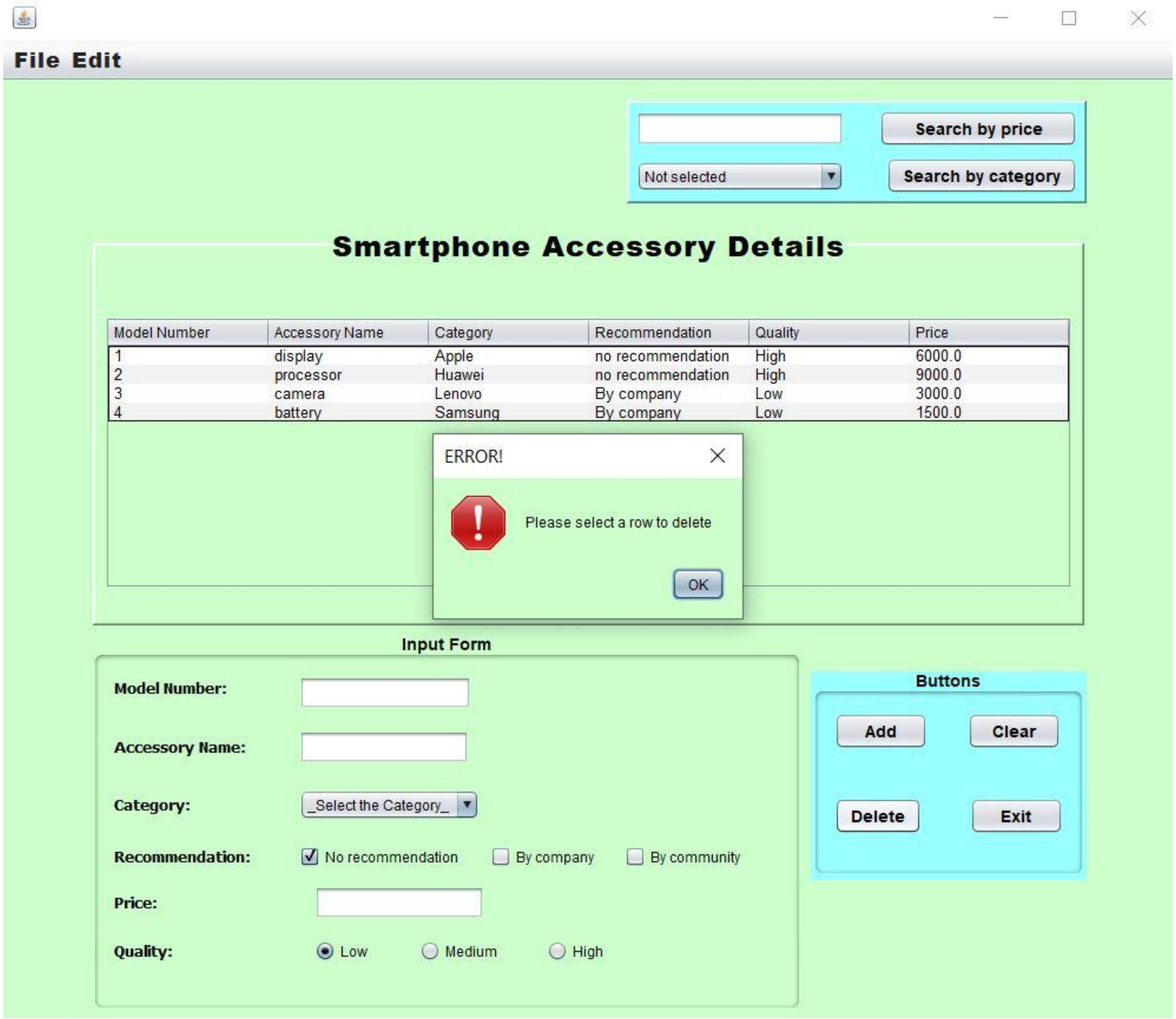
Category: _Select the Category_

Recommendation: ☑ No recommendation □ By company □ By community

Price:

Quality: ● Low ○ Medium ○ High

*Figure 31: click delete button*



| Model Number | Accessory Name | Category | Recommendation | Quality | Price |
|---|---|---|---|---|---|
| 1 | display | Apple | no recommendation | High | 6000.0 |
| 2 | processor | Huawei | no recommendation | High | 9000.0 |
| 3 | camera | Lenovo | By company | Low | 3000.0 |
| 4 | battery | Samsung | By company | Low | 1500.0 |

ERROR!

Please select a row to delete

Model Number:

Accessory Name:

Category: _Select the Category_

Recommendation: ☑ No recommendation □ By company □ By community

Price:

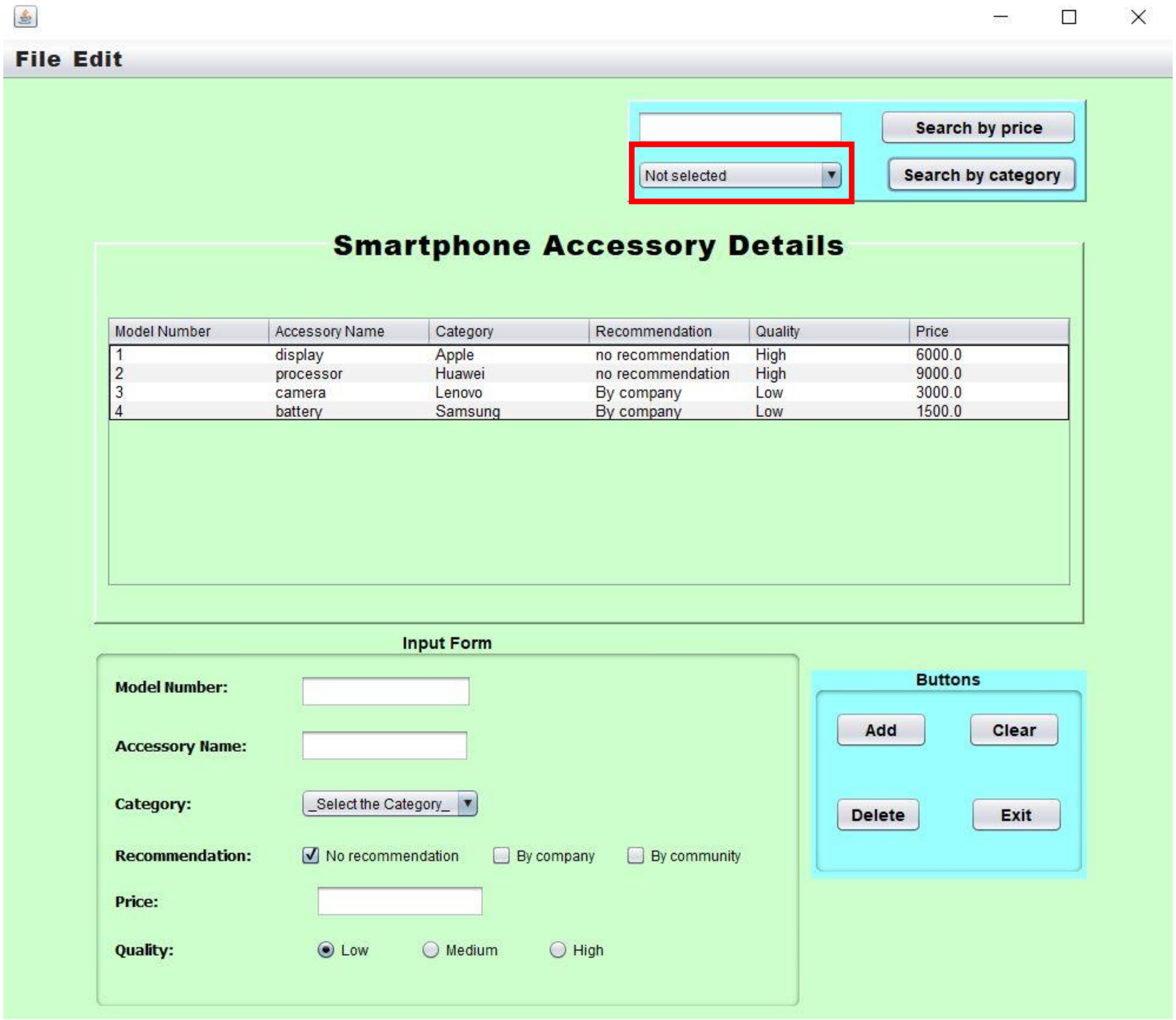
Quality: ● Low ○ Medium ○ High

*Figure 32: Display an error message*

### 5.3.5 Test C (5): Shows Dialog Box for not selecting the category item in search.

| | |
|---|---|
| **Objectives** | Display Dialog Box for not selecting the category Item in Search |
| **Action** | Not Selected Category Item<br>Click Search Button<br>Display Dialog Box “Please Select the Category” |
| **Expected Result** | Dialog Box will Display |
| **Actual Result** | Dialog Box is Display |
| **Conclusion** | Test was successful |

*Table 11: Test C (5): Shows Dialog Box for not selecting the category item in search.*



Not selected

| Model Number | Accessory Name | Category | Recommendation | Quality | Price |
|---|---|---|---|---|---|
| 1 | display | Apple | no recommendation | High | 6000.0 |
| 2 | processor | Huawei | no recommendation | High | 9000.0 |
| 3 | camera | Lenovo | By company | Low | 3000.0 |
| 4 | battery | Samsung | By company | Low | 1500.0 |

Model Number:

Accessory Name:

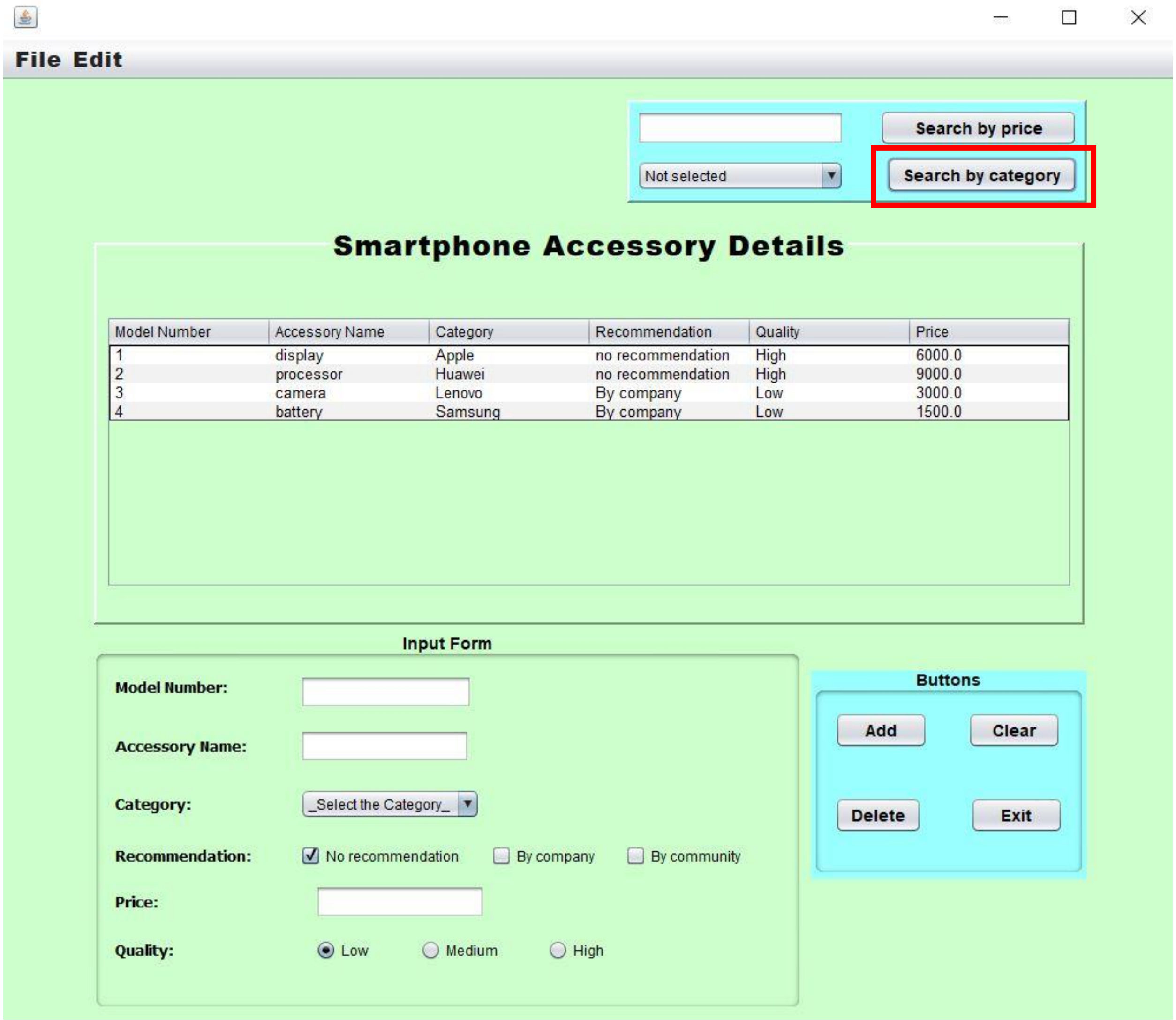
Category: _Select the Category_

Recommendation: ☑ No recommendation ☐ By company ☐ By community

Price:

Quality: ● Low ○ Medium ○ High

*Figure 33: let the category remain unselected*



Not selected

| Model Number | Accessory Name | Category | Recommendation | Quality | Price |
|---|---|---|---|---|---|
| 1 | display | Apple | no recommendation | High | 6000.0 |
| 2 | processor | Huawei | no recommendation | High | 9000.0 |
| 3 | camera | Lenovo | By company | Low | 3000.0 |
| 4 | battery | Samsung | By company | Low | 1500.0 |

Model Number:

Accessory Name:

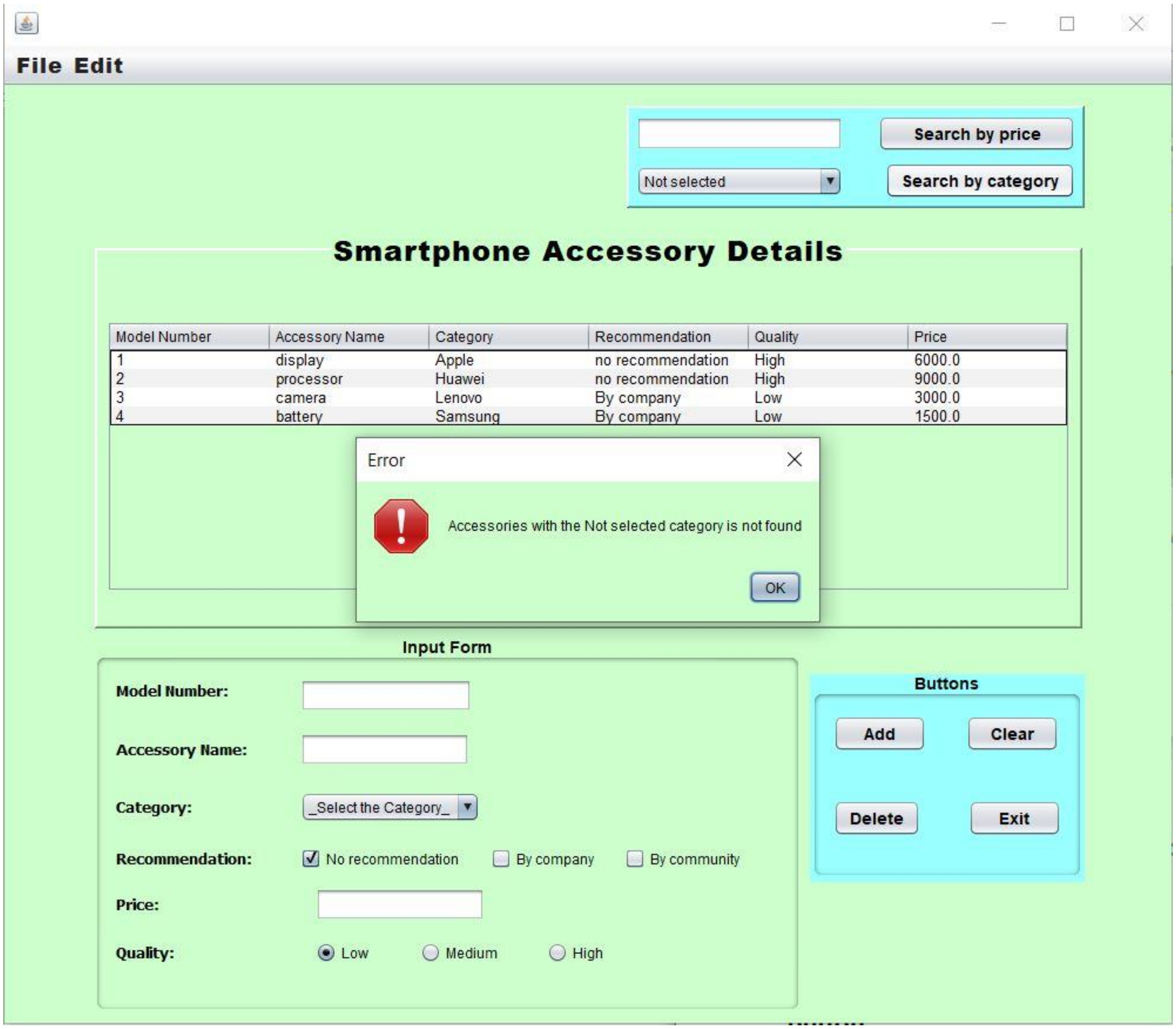
Category: _Select the Category_

Recommendation: ☑ No recommendation ☐ By company ☐ By community

Price:

Quality: ● Low ○ Medium ○ High

*Figure 34: click search button*



Smartphone Accessory Details

| Model Number | Accessory Name | Category | Recommendation | Quality | Price |
|---|---|---|---|---|---|
| 1 | display | Apple | no recommendation | High | 6000.0 |
| 2 | processor | Huawei | no recommendation | High | 9000.0 |
| 3 | camera | Lenovo | By company | Low | 3000.0 |
| 4 | battery | Samsung | By company | Low | 1500.0 |

Error

Accessories with the Not selected category is not found

OK

Model Number:

Accessory Name:

Category: _Select the Category_

Recommendation: ☑ No recommendation □ By company □ By community

Price:

Quality: ● Low ○ Medium ○ High

*Figure 35: displays error message*

## 6. Conclusion

“Emerging Programming Platforms and Technology” was a groupwork provided by London Metropolitan University and Itahari international College in the guidance of our module leader Mr. Pradhumna Dhungana. This groupwork was related with how can be deal in the real-time scenario for creating Information System while working in a team. Java Programming language and NetBeans IDE was the main components that are used for the completion of our “Optimus IS”.

In the coursework, we used various java swing items for successfully creating the user interface. The user would be able to store data in the table and query it according to price and category. For avoiding errors by the user the design was made simple and accessible for all type of users who have simple english knowledge. Help menu bar was created so that the user can grasp idea to use the system properly. In case of wrong commands by the user, error messages will be popped up so the mistake can be fixed without difficulties.

Working in a group as a team was the best part of this groupwork. This groupwork helped us to overcome our lack of teamwork and helped us to increase our social interaction with the group member. This groupwork taught us how to use the individual ability in a group while completing their shortcomings by other group members assistance and guidance. When working in a group, we get chance to share our ideas and as well as get to know the new ideas form other group members. While doing this coursework, we didn’t try to do anything all by ourselves. At first, we discuss about the topic in detail and took some ideas from our group members then only we proceeded to work on that topic.

As we all know that teamwork makes the dream work, we completed this groupwork of module ‘Emerging Programming Platforms and technologies’. Each member of the group gave their best in the completion of this groupwork. In some parts, where a group member got confused then other group members helped in understanding that topic. “Many hands make work easier”, as said in the proverb, doing the groupwork by dividing the task and helping each other in some difficult parts, this groupwork has been completed in easier way. If this groupwork was given as an individual work, then it wouldn’t have been as efficient as the way we completed this groupwork. While completing this groupwork, we helped each other to fill their incompleteness and flaws with the power of teamwork.

## 7. **References**

Bolton, D. (2019, July 03). *what-is-programming*. Retrieved from thoughtco.com: https://www.thoughtco.com/what-is-programming-958331

Computer Hope. (2020, Sept 06). *programming-language*. Retrieved from computerhope.com: https://www.computerhope.com/jargon/p/programming-language.htm

Geeks for Geeks. (2020, Aug 10). *algorithms-in-java*. Retrieved from geeksforgeeks.org: https://www.geeksforgeeks.org/searching-algorithms-in-java/

Parvez, F. (2020, Nov 10). *binary-search-algorithm*. Retrieved from mygreatlearning.com: https://www.mygreatlearning.com/blog/binary-search-algorithm/

Rishabh Prabhu. (2019, Feb 14). *java-programming-basics*. Retrieved from geeksforgeeks.org: https://www.geeksforgeeks.org/java-programming-basics/

# 8. Appendix

```
/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */
package AppliancesInfo;
import java.awt.Desktop;
import java.awt.event.KeyEvent;
import java.io.File;
import java.io.IOException;
import java.util.ArrayList;
import javax.swing.JOptionPane;
import javax.swing.table.DefaultTableModel;

/**
 *
 * @author Delish Khadka
 */
public class AppliancesInfo extends javax.swing.JFrame {
```

```
private ArrayList<PhoneDetails> list = new ArrayList();

private String modelNumber;

private String accessoryName;

private double price;

private String category;

private String quality;

private String recommendation;


/**

 * Creates new form AppliancesInfo

 */


public AppliancesInfo() {

    initComponents();

}


/**

 * This method is called from within the constructor to initialize the form.

 * WARNING: Do NOT modify this code. The content of this method is always

 * regenerated by the Form Editor.

 */

@SuppressWarnings("unchecked")

// <editor-fold defaultstate="collapsed" desc="Generated Code">
```

```
    private void initComponents() {

        buttonGroup1 = new javax.swing.ButtonGroup();
        buttonGroup2 = new javax.swing.ButtonGroup();
        tablePanel = new javax.swing.JPanel();
        jScrollPane1 = new javax.swing.JScrollPane();
        tblDetails = new javax.swing.JTable();
        searchPanel = new javax.swing.JPanel();
        searchCategoryBtn = new javax.swing.JButton();
        txtSearchByPrice = new javax.swing.JTextField();
        searchCategoryCmbBox = new javax.swing.JComboBox<>();
        searchBtn = new javax.swing.JButton();
        inputPanel = new javax.swing.JPanel();
        jLabel3 = new javax.swing.JLabel();
        jLabel6 = new javax.swing.JLabel();
        jLabel7 = new javax.swing.JLabel();
        txtAccName = new javax.swing.JTextField();
        txtModelNum = new javax.swing.JTextField();
        medRadioBtn = new javax.swing.JRadioButton();
        jLabel1 = new javax.swing.JLabel();
        highRadioBtn = new javax.swing.JRadioButton();
        txtPrice = new javax.swing.JTextField();
        comboCategory = new javax.swing.JComboBox<>();
```

```
compaCheckBox = new javax.swing.JCheckBox();

jLabel4 = new javax.swing.JLabel();

lowRadioBtn = new javax.swing.JRadioButton();

commuCheckBox = new javax.swing.JCheckBox();

jLabel2 = new javax.swing.JLabel();

norecoCheckBox = new javax.swing.JCheckBox();

buttonPanel = new javax.swing.JPanel();

addBtn = new javax.swing.JButton();

btnDelete = new javax.swing.JButton();

clearBtn = new javax.swing.JButton();

exitBtn = new javax.swing.JButton();

jMenuBar1 = new javax.swing.JMenuBar();

jMenu1 = new javax.swing.JMenu();

jMenuOpen = new javax.swing.JMenuItem();

jSeparator1 = new javax.swing.JPopupMenu.Separator();

jMenuExit = new javax.swing.JMenuItem();

jMenu2 = new javax.swing.JMenu();

jMenuHelp = new javax.swing.JMenuItem();


setDefaultCloseOperation(javax.swing.WindowConstants.EXIT_ON_CLOSE);


tablePanel.setBackground(new java.awt.Color(204, 255, 204));
tablePanel.setBorder(javax.swing.BorderFactory.createTitledBorder(javax.swing.BorderFactory.createBevelBorder(javax.swing.border.BevelBorder.RAISED), "Smartphone Accessory Details", javax.swing.border.TitledBorder.CENTER, javax.swing.border.TitledBorder.TOP, new java.awt.Font("Arial Black", 1, 24))); // NOI18N


        tblDetails.setBorder(javax.swing.BorderFactory.createLineBorder(new java.awt.Color(0, 0, 0)));
        tblDetails.setFont(new java.awt.Font("Arial", 0, 13)); // NOI18N
        tblDetails.setModel(new javax.swing.table.DefaultTableModel(
            new Object [][] {


            },
            new String [] {
                "Model Number", "Accessory Name", "Category", "Recommendation", "Quality", "Price"
            }
        ) {
            boolean[] canEdit = new boolean [] {
                false, false, false, false, false, false
            };


            public boolean isCellEditable(int rowIndex, int columnIndex) {
                return canEdit [columnIndex];
```

```
      }
    });
    tblDetails.setGridColor(new java.awt.Color(204, 204, 0));
    tblDetails.getTableHeader().setReorderingAllowed(false);
    jScrollPane1.setViewportView(tblDetails);

    javax.swing.GroupLayout tablePanelLayout = new
javax.swing.GroupLayout(tablePanel);
    tablePanel.setLayout(tablePanelLayout);
    tablePanelLayout.setHorizontalGroup(

tablePanelLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
      .addGroup(tablePanelLayout.createSequentialGroup()
        .addContainerGap()
        .addComponent(jScrollPane1)
        .addContainerGap())
    );
    tablePanelLayout.setVerticalGroup(

tablePanelLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
      .addGroup(javax.swing.GroupLayout.Alignment.TRAILING,
tablePanelLayout.createSequentialGroup()
        .addContainerGap(41, Short.MAX_VALUE)
```

```
                .addComponent(jScrollPane1,  javax.swing.GroupLayout.PREFERRED_SIZE, 231, javax.swing.GroupLayout.PREFERRED_SIZE)
                .addGap(27, 27, 27))
        );

        searchPanel.setBackground(new java.awt.Color(153, 255, 255));

searchPanel.setBorder(javax.swing.BorderFactory.createBevelBorder(javax.swing.border.BevelBorder.RAISED));

        searchCategoryBtn.setFont(new java.awt.Font("SansSerif", 1, 14)); // NOI18N
        searchCategoryBtn.setText("Search by category");
        searchCategoryBtn.addActionListener(new java.awt.event.ActionListener() {
            public void actionPerformed(java.awt.event.ActionEvent evt) {
                searchCategoryBtnActionPerformed(evt);
            }
        });

        txtSearchByPrice.setFont(new java.awt.Font("Tahoma", 0, 14)); // NOI18N
        txtSearchByPrice.addActionListener(new java.awt.event.ActionListener() {
            public void actionPerformed(java.awt.event.ActionEvent evt) {
                txtSearchByPriceActionPerformed(evt);
            }
        });
        txtSearchByPrice.addKeyListener(new java.awt.event.KeyAdapter() {
            public void keyTyped(java.awt.event.KeyEvent evt) {
                txtSearchByPriceKeyTyped(evt);
            }
        });

        searchCategoryCmbBox.setModel(new javax.swing.DefaultComboBoxModel<>(new String[] { "Not selected", "Apple", "Huawei", "Lenovo", "Samsung", "Xiaomi" }));

        searchBtn.setFont(new java.awt.Font("SansSerif", 1, 14)); // NOI18N
        searchBtn.setText("Search by price");
        searchBtn.addActionListener(new java.awt.event.ActionListener() {
            public void actionPerformed(java.awt.event.ActionEvent evt) {
                searchBtnActionPerformed(evt);
            }
        });

        javax.swing.GroupLayout searchPanelLayout = new javax.swing.GroupLayout(searchPanel);
        searchPanel.setLayout(searchPanelLayout);
        searchPanelLayout.setHorizontalGroup(
searchPanelLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING
)
            .addGroup(searchPanelLayout.createSequentialGroup()
                .addContainerGap(javax.swing.GroupLayout.DEFAULT_SIZE,
Short.MAX_VALUE)

.addGroup(searchPanelLayout.createParallelGroup(javax.swing.GroupLayout.Alignmen
t.TRAILING, false)
                    .addGroup(searchPanelLayout.createSequentialGroup()
                        .addComponent(txtSearchByPrice,
javax.swing.GroupLayout.PREFERRED_SIZE, 176,
javax.swing.GroupLayout.PREFERRED_SIZE)
                        .addGap(30, 30, 30)
                        .addComponent(searchBtn,
javax.swing.GroupLayout.PREFERRED_SIZE, 168,
javax.swing.GroupLayout.PREFERRED_SIZE))
                    .addGroup(searchPanelLayout.createSequentialGroup()
                        .addComponent(searchCategoryCmbBox,
javax.swing.GroupLayout.PREFERRED_SIZE, 176,
javax.swing.GroupLayout.PREFERRED_SIZE)

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED,
javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)
                        .addComponent(searchCategoryBtn)))
                .addContainerGap())
```

```
    );
    searchPanelLayout.setVerticalGroup(

searchPanelLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING
)
      .addGroup(searchPanelLayout.createSequentialGroup()
        .addContainerGap()

.addGroup(searchPanelLayout.createParallelGroup(javax.swing.GroupLayout.Alignmen
t.BASELINE)
          .addComponent(txtSearchByPrice,
javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE)
          .addComponent(searchBtn))
        .addGap(9, 9, 9)

.addGroup(searchPanelLayout.createParallelGroup(javax.swing.GroupLayout.Alignmen
t.BASELINE)
          .addComponent(searchCategoryBtn)
          .addComponent(searchCategoryCmbBox,
javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE))
        .addContainerGap())
    );
    inputPanel.setBackground(new java.awt.Color(204, 255, 204));
    inputPanel.setBorder(javax.swing.BorderFactory.createTitledBorder(null, "Input Form", javax.swing.border.TitledBorder.CENTER, javax.swing.border.TitledBorder.DEFAULT_POSITION, new java.awt.Font("Arial", 1, 14))); // NOI18N

    jLabel3.setFont(new java.awt.Font("Tahoma", 1, 13)); // NOI18N
    jLabel3.setText("Accessory Name:");

    jLabel6.setFont(new java.awt.Font("Tahoma", 1, 13)); // NOI18N
    jLabel6.setText("Price:");

    jLabel7.setFont(new java.awt.Font("Tahoma", 1, 13)); // NOI18N
    jLabel7.setText("Quality:");

    txtAccName.addActionListener(new java.awt.event.ActionListener() {
        public void actionPerformed(java.awt.event.ActionEvent evt) {
            txtAccNameActionPerformed(evt);
        }
    });

    txtModelNum.addActionListener(new java.awt.event.ActionListener() {
```

```
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        txtModelNumActionPerformed(evt);
    }
});
txtModelNum.addKeyListener(new java.awt.event.KeyAdapter() {
    public void keyTyped(java.awt.event.KeyEvent evt) {
        txtModelNumKeyTyped(evt);
    }
});

buttonGroup1.add(medRadioBtn);
medRadioBtn.setText("Medium");
medRadioBtn.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        medRadioBtnActionPerformed(evt);
    }
});

jLabel1.setFont(new java.awt.Font("Tahoma", 1, 13)); // NOI18N
jLabel1.setText("Model Number:");

buttonGroup1.add(highRadioBtn);
highRadioBtn.setText("High");
```

```
    highRadioBtn.addActionListener(new java.awt.event.ActionListener() {
        public void actionPerformed(java.awt.event.ActionEvent evt) {
            highRadioBtnActionPerformed(evt);
        }
    });

    txtPrice.addActionListener(new java.awt.event.ActionListener() {
        public void actionPerformed(java.awt.event.ActionEvent evt) {
            txtPriceActionPerformed(evt);
        }
    });
    txtPrice.addKeyListener(new java.awt.event.KeyAdapter() {
        public void keyTyped(java.awt.event.KeyEvent evt) {
            txtPriceKeyTyped(evt);
        }
    });

    comboCategory.setModel(new javax.swing.DefaultComboBoxModel<>(new String[]
{ "_Select the Category_", "Apple", "Huawei", "Lenovo", "Samsung", "Xiaomi" }));
    comboCategory.addActionListener(new java.awt.event.ActionListener() {
        public void actionPerformed(java.awt.event.ActionEvent evt) {
            comboCategoryActionPerformed(evt);
        }
    });

    buttonGroup2.add(compaCheckBox);
    compaCheckBox.setText("By company");
    compaCheckBox.addActionListener(new java.awt.event.ActionListener() {
        public void actionPerformed(java.awt.event.ActionEvent evt) {
            compaCheckBoxActionPerformed(evt);
        }
    });

    jLabel4.setFont(new java.awt.Font("Tahoma", 1, 13)); // NOI18N
    jLabel4.setText("Recommendation:");

    buttonGroup1.add(lowRadioBtn);
    lowRadioBtn.setSelected(true);
    lowRadioBtn.setText("Low");
    lowRadioBtn.addActionListener(new java.awt.event.ActionListener() {
        public void actionPerformed(java.awt.event.ActionEvent evt) {
            lowRadioBtnActionPerformed(evt);
        }
    });

    buttonGroup2.add(commuCheckBox);
```

```
commuCheckBox.setText("By community");
commuCheckBox.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        commuCheckBoxActionPerformed(evt);
    }
});

jLabel2.setFont(new java.awt.Font("Tahoma", 1, 13)); // NOI18N
jLabel2.setText("Category:");

buttonGroup2.add(norecoCheckBox);
norecoCheckBox.setSelected(true);
norecoCheckBox.setText("No recommendation");
norecoCheckBox.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        norecoCheckBoxActionPerformed(evt);
    }
});

javax.swing.GroupLayout inputPanelLayout = new javax.swing.GroupLayout(inputPanel);
inputPanel.setLayout(inputPanelLayout);
inputPanelLayout.setHorizontalGroup(
inputPanelLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
      .addGroup(inputPanelLayout.createSequentialGroup()
        .addContainerGap()

.addGroup(inputPanelLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.
LEADING)
          .addComponent(jLabel2)
          .addGroup(inputPanelLayout.createSequentialGroup()

.addGroup(inputPanelLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.
LEADING)

.addGroup(inputPanelLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.
LEADING)
                .addGroup(javax.swing.GroupLayout.Alignment.TRAILING,
inputPanelLayout.createSequentialGroup()
                  .addComponent(jLabel3)
                  .addGap(45, 45, 45))
                .addGroup(inputPanelLayout.createSequentialGroup()
                  .addComponent(jLabel1)
                  .addGap(61, 61, 61)))
              .addGroup(inputPanelLayout.createSequentialGroup()
                .addComponent(jLabel4)
                .addGap(41, 41, 41)))
```

```
.addGroup(inputPanelLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.
LEADING)
                        .addComponent(txtModelNum,
javax.swing.GroupLayout.PREFERRED_SIZE,                                        146,
javax.swing.GroupLayout.PREFERRED_SIZE)
                        .addComponent(txtAccName,
javax.swing.GroupLayout.PREFERRED_SIZE,                                        144,
javax.swing.GroupLayout.PREFERRED_SIZE)
                        .addGroup(inputPanelLayout.createSequentialGroup()
                            .addComponent(norecoCheckBox)
                            .addGap(27, 27, 27)
                            .addComponent(compaCheckBox)
                            .addGap(26, 26, 26)
                            .addComponent(commuCheckBox))
                        .addComponent(comboCategory,
javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE)))
                .addGroup(inputPanelLayout.createSequentialGroup()

.addGroup(inputPanelLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.
LEADING)
                        .addComponent(jLabel6)
                        .addComponent(jLabel7))
                    .addGap(120, 120, 120)
```

```
.addGroup(inputPanelLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.
LEADING)
              .addGroup(inputPanelLayout.createSequentialGroup()
                .addComponent(lowRadioBtn)
                .addGap(44, 44, 44)
                .addComponent(medRadioBtn)
                .addGap(42, 42, 42)
                .addComponent(highRadioBtn))
              .addComponent(txtPrice,
javax.swing.GroupLayout.PREFERRED_SIZE,                                    144,
javax.swing.GroupLayout.PREFERRED_SIZE))))
        .addContainerGap(40, Short.MAX_VALUE))
    );
    inputPanelLayout.setVerticalGroup(

inputPanelLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
      .addGroup(inputPanelLayout.createSequentialGroup()
        .addContainerGap()

.addGroup(inputPanelLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.
LEADING)
          .addGroup(inputPanelLayout.createSequentialGroup()
            .addGap(2, 2, 2)
            .addComponent(jLabel1))
          .addGroup(inputPanelLayout.createSequentialGroup()
            .addComponent(txtModelNum,
javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE)
            .addGap(18, 18, 18)

.addGroup(inputPanelLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.
BASELINE)
              .addComponent(jLabel3)
              .addComponent(txtAccName,
javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE))))
        .addGap(22, 22, 22)

.addGroup(inputPanelLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.
BASELINE)
          .addComponent(jLabel2)
          .addComponent(comboCategory,
javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE))
        .addGap(22, 22, 22)

.addGroup(inputPanelLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.
BASELINE)
```

```
                    .addComponent(jLabel4)
                    .addComponent(norecoCheckBox)
                    .addComponent(compaCheckBox)
                    .addComponent(commuCheckBox))
                .addGap(16, 16, 16)

.addGroup(inputPanelLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.
BASELINE)
                    .addComponent(jLabel6)
                    .addComponent(txtPrice,      javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE))
                .addGap(18, 18, 18)

.addGroup(inputPanelLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.
BASELINE)
                    .addComponent(medRadioBtn)
                    .addComponent(highRadioBtn)
                    .addComponent(lowRadioBtn)
                    .addComponent(jLabel7))
                .addContainerGap(31, Short.MAX_VALUE))
        );


        buttonPanel.setBackground(new java.awt.Color(153, 255, 255));
        buttonPanel.setBorder(javax.swing.BorderFactory.createTitledBorder(null,
"Buttons", javax.swing.border.TitledBorder.CENTER,
javax.swing.border.TitledBorder.DEFAULT_POSITION, new java.awt.Font("Arial", 1,
14))); // NOI18N


        addBtn.setFont(new java.awt.Font("SansSerif", 1, 14)); // NOI18N
        addBtn.setText("Add");
        addBtn.addActionListener(new java.awt.event.ActionListener() {
            public void actionPerformed(java.awt.event.ActionEvent evt) {
                addBtnActionPerformed(evt);
            }
        });

        btnDelete.setFont(new java.awt.Font("SansSerif", 1, 14)); // NOI18N
        btnDelete.setText("Delete");
        btnDelete.addActionListener(new java.awt.event.ActionListener() {
            public void actionPerformed(java.awt.event.ActionEvent evt) {
                btnDeleteActionPerformed(evt);
            }
        });

        clearBtn.setFont(new java.awt.Font("SansSerif", 1, 14)); // NOI18N
        clearBtn.setText("Clear");
```

```
    clearBtn.addActionListener(new java.awt.event.ActionListener() {
      public void actionPerformed(java.awt.event.ActionEvent evt) {
        clearBtnActionPerformed(evt);
      }
    });

    exitBtn.setFont(new java.awt.Font("SansSerif", 1, 14)); // NOI18N
    exitBtn.setText("Exit");
    exitBtn.addActionListener(new java.awt.event.ActionListener() {
      public void actionPerformed(java.awt.event.ActionEvent evt) {
        exitBtnActionPerformed(evt);
      }
    });

    javax.swing.GroupLayout buttonPanelLayout = new
javax.swing.GroupLayout(buttonPanel);
    buttonPanel.setLayout(buttonPanelLayout);
    buttonPanelLayout.setHorizontalGroup(

buttonPanelLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
      .addGroup(buttonPanelLayout.createSequentialGroup()
        .addContainerGap()
```

```
.addGroup(buttonPanelLayout.createParallelGroup(javax.swing.GroupLayout.Alignment
.LEADING)
               .addGroup(buttonPanelLayout.createSequentialGroup()
                  .addComponent(addBtn, javax.swing.GroupLayout.PREFERRED_SIZE,
79, javax.swing.GroupLayout.PREFERRED_SIZE)
                  .addGap(34, 34, 34)
                  .addComponent(clearBtn,
javax.swing.GroupLayout.PREFERRED_SIZE, 79,
javax.swing.GroupLayout.PREFERRED_SIZE)
                  .addGap(0, 2, Short.MAX_VALUE))
               .addGroup(buttonPanelLayout.createSequentialGroup()
                  .addComponent(btnDelete)

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED,
javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)
                  .addComponent(exitBtn, javax.swing.GroupLayout.PREFERRED_SIZE,
79, javax.swing.GroupLayout.PREFERRED_SIZE)))
            .addContainerGap())
      );
      buttonPanelLayout.setVerticalGroup(

buttonPanelLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
         .addGroup(buttonPanelLayout.createSequentialGroup()
            .addContainerGap()
```

```
.addGroup(buttonPanelLayout.createParallelGroup(javax.swing.GroupLayout.Alignment
.BASELINE)
          .addComponent(addBtn)
          .addComponent(clearBtn))
        .addGap(41, 41, 41)

.addGroup(buttonPanelLayout.createParallelGroup(javax.swing.GroupLayout.Alignment
.BASELINE)
          .addComponent(btnDelete)
          .addComponent(exitBtn))
        .addContainerGap(25, Short.MAX_VALUE))
    );


    jMenu1.setText("File");
    jMenu1.setFont(new java.awt.Font("Arial Black", 1, 18)); // NOI18N
    jMenu1.addActionListener(new java.awt.event.ActionListener() {
      public void actionPerformed(java.awt.event.ActionEvent evt) {
        jMenu1ActionPerformed(evt);
      }
    });
```

```
jMenuOpen.setAccelerator(javax.swing.KeyStroke.getKeyStroke(java.awt.event.KeyEvent.VK_O, java.awt.event.InputEvent.CTRL_MASK));
        jMenuOpen.setText("Open");
        jMenuOpen.addActionListener(new java.awt.event.ActionListener() {
            public void actionPerformed(java.awt.event.ActionEvent evt) {
                jMenuOpenActionPerformed(evt);
            }
        });
        jMenu1.add(jMenuOpen);
        jMenu1.add(jSeparator1);


jMenuExit.setAccelerator(javax.swing.KeyStroke.getKeyStroke(java.awt.event.KeyEvent.VK_C, java.awt.event.InputEvent.ALT_MASK | java.awt.event.InputEvent.CTRL_MASK));
        jMenuExit.setText("Exit");
        jMenuExit.addActionListener(new java.awt.event.ActionListener() {
            public void actionPerformed(java.awt.event.ActionEvent evt) {
                jMenuExitActionPerformed(evt);
            }
        });
        jMenu1.add(jMenuExit);
        jMenuBar1.add(jMenu1);


        jMenu2.setText("Edit");
        jMenu2.setFont(new java.awt.Font("Arial Black", 1, 18)); // NOI18N


jMenuHelp.setAccelerator(javax.swing.KeyStroke.getKeyStroke(java.awt.event.KeyEvent.VK_H, java.awt.event.InputEvent.ALT_MASK | java.awt.event.InputEvent.CTRL_MASK));
        jMenuHelp.setText("Help");
        jMenuHelp.addActionListener(new java.awt.event.ActionListener() {
            public void actionPerformed(java.awt.event.ActionEvent evt) {
                jMenuHelpActionPerformed(evt);
            }
        });
        jMenu2.add(jMenuHelp);

        jMenuBar1.add(jMenu2);

        setJMenuBar(jMenuBar1);

        javax.swing.GroupLayout layout = new javax.swing.GroupLayout(getContentPane());
        getContentPane().setLayout(layout);
```

```
    layout.setHorizontalGroup(
      layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
      .addGroup(javax.swing.GroupLayout.Alignment.TRAILING,
layout.createSequentialGroup()

.addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.TRAILING)
          .addGroup(layout.createSequentialGroup()
            .addContainerGap(javax.swing.GroupLayout.DEFAULT_SIZE,
Short.MAX_VALUE)
            .addComponent(searchPanel,
javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE))
          .addGroup(layout.createSequentialGroup()
            .addGap(75, 75, 75)

.addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
              .addComponent(tablePanel,
javax.swing.GroupLayout.DEFAULT_SIZE,   javax.swing.GroupLayout.DEFAULT_SIZE,
Short.MAX_VALUE)
              .addGroup(layout.createSequentialGroup()
                .addComponent(inputPanel,
javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE)
.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
                                .addComponent(buttonPanel,
javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE,
Short.MAX_VALUE)))))
                .addGap(74, 74, 74))
        );
        layout.setVerticalGroup(
            layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
            .addGroup(javax.swing.GroupLayout.Alignment.TRAILING,
layout.createSequentialGroup()
                .addGap(18, 18, 18)
                .addComponent(searchPanel, javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)
                .addGap(18, 18, 18)
                .addComponent(tablePanel, javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE)

.addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
                    .addGroup(layout.createSequentialGroup()
                        .addGap(37, 37, 37)
                        .addComponent(buttonPanel,
javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE))
```

```
                    .addGroup(layout.createSequentialGroup()
                        .addGap(6, 6, 6)
                        .addComponent(inputPanel,
javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE)))
                .addContainerGap(javax.swing.GroupLayout.DEFAULT_SIZE,
Short.MAX_VALUE))
        );

        pack();
    }// </editor-fold>

    private void txtModelNumActionPerformed(java.awt.event.ActionEvent evt) {
        // TODO add your handling code here:
    }

    private void txtPriceActionPerformed(java.awt.event.ActionEvent evt) {
        // TODO add your handling code here:
    }

    private void txtAccNameActionPerformed(java.awt.event.ActionEvent evt) {
        // TODO add your handling code here:
    }
    /*
    This method depends on the PhoneDetails class
    This method is used when add button is pressed after inputting values in text field and selecting other items
    This table adds the data entered and show it in the JTable form
    In case of wrong input by the user, error message is popped with proper messages to fix the error
    */
    private void addBtnActionPerformed(java.awt.event.ActionEvent evt) {
        // TODO add your handling code here:
        try
                {
                        modelNumber = txtModelNum.getText().trim();
                        accessoryName = txtAccName.getText().trim();
                        category = comboCategory.getSelectedItem().toString();

                        if((modelNumber.isEmpty() || accessoryName.isEmpty()))
                  {
                      JOptionPane.showMessageDialog(rootPane, "Please enter details in all fieds!", " Invalid", JOptionPane.ERROR_MESSAGE);

                  }
```

```
else if(comboCategory.getSelectedIndex() == 0){
       JOptionPane.showMessageDialog(rootPane, "Please select a category!","Error",2);
     return;
    }


        else{
    price = Double.parseDouble(txtPrice.getText().trim());
    if(price<=0){
        JOptionPane.showMessageDialog(rootPane, "Price cannot be zero", " Invalid", JOptionPane.ERROR_MESSAGE);
        return;
    }
    if(lowRadioBtn.isSelected())
    {
        quality=lowRadioBtn.getText();
    }
    else if(medRadioBtn.isSelected())
    {
        quality=medRadioBtn.getText();
    }
    else
    {
            quality=highRadioBtn.getText();
        }


        if(norecoCheckBox.isSelected())
        {
            recommendation = "no recommendation";
        }
        else if(compaCheckBox.isSelected())
        {
            recommendation = "By company";
        }
        else if(commuCheckBox.isSelected())
        {
            recommendation = "By community";
        }
        if(validation())
        {
            DefaultTableModel model = (DefaultTableModel)
tblDetails.getModel();
            model.addRow(new Object[] { modelNumber, accessoryName,
category, recommendation, quality, price});
            PhoneDetails phoneDetails = new PhoneDetails( modelNumber,
accessoryName, category, recommendation, quality, price);//creating object of
PhoneDetails class
```

```
                        list.add(phoneDetails);
                    }
                 }
               }
               catch(NumberFormatException exception){
                        JOptionPane.showMessageDialog(rootPane, "Please enter value in
price text field!","Invalid Information",JOptionPane.ERROR_MESSAGE);
               }
   }


   /*
   This method is used to validate user inputs
   In case of duplicate input in model number and accessories name, error message is
shown to the user
   If user input is validated, returns true , returns false if the data is same on two entries
   */
   public boolean validation(){
        if(list.isEmpty()){
               return true;
        }
        for(PhoneDetails ArrayList:list)
        {
        if(ArrayList.getModelNumber().equals(modelNumber))
```

```
    {
    JOptionPane.showMessageDialog(rootPane, "The Model Number is already used in the table!", "Refill information!", JOptionPane.ERROR_MESSAGE);
        return false;
    }
    else if(!ArrayList.getQuality().equals(quality)){



    if(ArrayList.getAccessoryName().toLowerCase().equals(accessoryName.toLowerCase())){
        return true;
        }
    }


    else if(ArrayList.getAccessoryName().toLowerCase().equals(accessoryName.toLowerCase())){
        JOptionPane.showMessageDialog(rootPane, "Accessory name cannot be same of two entries!", "Refill information!", JOptionPane.ERROR_MESSAGE);
        return false;
    }
    }
     return true;
    }
    /*
    This button is used for functioning exit button
    Pop up is created so that user can choose to exit, in case of mistakenly exit
    */
    private void exitBtnActionPerformed(java.awt.event.ActionEvent evt) {
        // TODO add your handling code here:
        //System.exit(0);
        int confirmed = JOptionPane.showConfirmDialog(null, "Are you sure, you want to Exit Program?","EXIT",JOptionPane.YES_NO_OPTION);
        if(confirmed == JOptionPane.YES_OPTION)
        {
            dispose();
        }
    }

    private void comboCategoryActionPerformed(java.awt.event.ActionEvent evt) {
        // TODO add your handling code here:
    }

    /*
    This button clears all the textfields in the GUI and resets the value in input form to default
    */
    private void clearBtnActionPerformed(java.awt.event.ActionEvent evt) {
        // TODO add your handling code here:
        txtModelNum.setText("");
        txtAccName.setText("");
        txtPrice.setText("");
        comboCategory.setSelectedIndex(0);
        norecoCheckBox.setSelected(true);
        lowRadioBtn.setSelected(true);
        JOptionPane.showMessageDialog(rootPane, "Cleared!", "Info!", JOptionPane.INFORMATION_MESSAGE);
    }

    /*
    This method is used to search accessories according to category from combobox

    */
    private void searchCategoryBtnActionPerformed(java.awt.event.ActionEvent evt) {
        // TODO add your handling code here:
        if(list.isEmpty()){
                        JOptionPane.showMessageDialog(rootPane,"No items found for searching!","Message",JOptionPane.INFORMATION_MESSAGE);
                return;
        }
        String searchByCategory =
searchCategoryCmbBox.getSelectedItem().toString();
        String accessoryName = "";
        String details = "";
        int count = 0;
        for(PhoneDetails phoneDetails:list){
        if(phoneDetails.getCategory().equals(searchByCategory)){
            count++;
            accessoryName += count+ ". The Accessory Name is
"+phoneDetails.getAccessoryName()+".\n";
            details += "The Model Number is
"+phoneDetails.getModelNumber()+",\n"
            + "The Accessory name is
"+phoneDetails.getAccessoryName()+",\n"
            + "The Category is "+phoneDetails.getCategory()+",\n"
            + "The Recommendation is
"+phoneDetails.getRecommendation()+",\n"
            + "The Quality is "+phoneDetails.getQuality()+" and\n"
            + "The Price is " +phoneDetails.getPrice() +" \n \n";
            }
        }
        if(count == 0){
                JOptionPane.showMessageDialog(rootPane,"Accessories with the
"+searchByCategory+" category is not
found","Error",JOptionPane.ERROR_MESSAGE);

        } else if(count==1){

                JOptionPane.showMessageDialog(rootPane,"One accessory with
the category
"+category+"found"+"\n"+details,"Message",JOptionPane.INFORMATION_MESSAGE);

        }

        else{

                int input = JOptionPane.showConfirmDialog(rootPane,"There are
"+count+" accessories found: \n"

                +accessoryName

                + "Do you want to view the details of the Accessories
found?","MESSAGE",JOptionPane.YES_NO_OPTION,
JOptionPane.INFORMATION_MESSAGE);

                if(input == 0){//yes


    JOptionPane.showMessageDialog(rootPane,details,"Accessory
Found",JOptionPane.INFORMATION_MESSAGE);

                }

        }



 }


 private void txtPriceKeyTyped(java.awt.event.KeyEvent evt) {
        // TODO add your handling code here:
        char key = evt.getKeyChar();
        if (!((key >= '0') && (key<='9') || key==KeyEvent.VK_BACK_SPACE || key==KeyEvent.VK_DELETE)) {
            evt.consume();
        }


    }

    private void txtModelNumKeyTyped(java.awt.event.KeyEvent evt) {
        // TODO add your handling code here:
        char key = evt.getKeyChar();
        if (!((key >= '0') && (key<='9') || key==KeyEvent.VK_BACK_SPACE || key==KeyEvent.VK_DELETE)) {
            evt.consume();
        }
    }

    private void txtSearchByPriceKeyTyped(java.awt.event.KeyEvent evt) {
        // TODO add your handling code here:
    }

    private void lowRadioBtnActionPerformed(java.awt.event.ActionEvent evt) {
        // TODO add your handling code here:

    }

    private void medRadioBtnActionPerformed(java.awt.event.ActionEvent evt) {
        // TODO add your handling code here:

    }

    private void highRadioBtnActionPerformed(java.awt.event.ActionEvent evt) {
        // TODO add your handling code here:

    }

    private void norecoCheckBoxActionPerformed(java.awt.event.ActionEvent evt) {
        // TODO add your handling code here:

    }

    private void compaCheckBoxActionPerformed(java.awt.event.ActionEvent evt) {
        // TODO add your handling code here:

    }
    private void commuCheckBoxActionPerformed(java.awt.event.ActionEvent evt) {
        // TODO add your handling code here:


    }


    /*
    This button deletes the selected row in the table
    */
    private void btnDeleteActionPerformed(java.awt.event.ActionEvent evt) {
        // TODO add your handling code here:
        try{DefaultTableModel model = (DefaultTableModel) tblDetails.getModel();
                int selectedRowIndex = tblDetails.getSelectedRow();
                if(model.getRowCount()==0){
                        JOptionPane.showMessageDialog(rootPane,"No rows found for deletion","ERROR!",JOptionPane.ERROR_MESSAGE);
                }else if(selectedRowIndex == -1){
                        JOptionPane.showMessageDialog(rootPane,"Please select a row to delete","ERROR!",JOptionPane.ERROR_MESSAGE);
                }
                else{
                        model.removeRow(selectedRowIndex);
                        list.remove(selectedRowIndex);
                }
            }
                catch(Exception e)
                {JOptionPane.showMessageDialog(rootPane,"Row deleted successfully","Message!",JOptionPane.INFORMATION_MESSAGE);
                }


    }


    private void jMenuOpenActionPerformed(java.awt.event.ActionEvent evt) {
        // TODO add your handling code here:
        try{
            File file = new File("Table.xlsx");
            if(file.exists()){
                if(Desktop.isDesktopSupported()){
                    Desktop.getDesktop().open(file);
                }
            }else{
                    JOptionPane.showMessageDialog(rootPane, "File could not be found", "Error", JOptionPane.ERROR_MESSAGE);
            }
            }catch(Exception e){
                    e.printStackTrace();
        }


    }


    private void jMenu1ActionPerformed(java.awt.event.ActionEvent evt) {
        // TODO add your handling code here:
    }


    private void jMenuExitActionPerformed(java.awt.event.ActionEvent evt) {
        // TODO add your handling code here:
        int confirmed = JOptionPane.showConfirmDialog(null, "Are you sure, you want to
Exit Program?","EXIT",JOptionPane.YES_NO_OPTION);
        if(confirmed == JOptionPane.YES_OPTION)
        {
            dispose();
        }


    }


    private void jMenuHelpActionPerformed(java.awt.event.ActionEvent evt) {
        // TODO add your handling code here:
        try{
            File help = new File("ReadMe.txt");
```

```
          if(help.exists()){
             if(Desktop.isDesktopSupported()){
                Desktop.getDesktop().open(help);
             }
          }else{
             JOptionPane.showMessageDialog(rootPane, "File could not be Found",
"Error", JOptionPane.ERROR_MESSAGE);
          }
       }catch(IOException e){


       }
    }


    /*
    This method re-arranges the sorted list as default
    This method is called in search by price method
    */
    private void toList(){
         list.clear();
         DefaultTableModel model = (DefaultTableModel) tblDetails.getModel();
         for(int i=0; i<model.getRowCount(); i++){
          String numList = model.getValueAt(i, 0).toString();
          String nameList = model.getValueAt(i, 1).toString();
```

```
            String categoryList = model.getValueAt(i, 2).toString();

            String recommendationList = model.getValueAt(i, 3).toString();

            String qualityList = model.getValueAt(i, 4).toString();

            double priceList = Double.parseDouble(model.getValueAt(i, 5).toString());

            list.add(new PhoneDetails(numList, nameList, categoryList, recommendationList,
qualityList, priceList));

          }

    }


    /*

    This method finds the index of list where the value is located

    start is the index from where the search begins

    end is the index upto which the search is done

    value is the price to perform search

    */

    private int binarySearch(ArrayList<PhoneDetails> list, int start, int end, double value){

                if(start<=end){

                int mid = (end + start)/2;

                        if(list.get(mid).getPrice()== value){

                                return mid;

                        }

                        else if(list.get(mid).getPrice()< value){

                                return binarySearch(list, mid+1, end, value);
                        }else{
                                return binarySearch(list,start, mid-1, value);
                        }
                        }else{
                                return -1;//returns -1 if the value is not found
                        }
   }
   /*
   This method sorts unsorted list
   */
   private void sort(ArrayList<PhoneDetails> list)
   {
        for(int i=0;i<list.size()-1;i++){
         int minPosition = minimumPosition(list,i);//calling minimumPosition method to find
smallest value in list
         PhoneDetails phoneDetails = list.get(i);
         list.set(i,list.get(minPosition));
         list.set(minPosition, phoneDetails);
      }
   }


   /*
   This method compares the value of the index with other indexes
   */
   private int minimumPosition(ArrayList<PhoneDetails> list,int x){//x = index of the list
        int minPosition= x;
      for(int i = x+1;i<list.size();i++)
        {
         if(list.get(i).getPrice() < list.get(minPosition).getPrice()){
         minPosition = i;
         }
        }
        return minPosition;
   }


   /*
   This method searches the accessories according to price
   */
   private void searchBtnActionPerformed(java.awt.event.ActionEvent evt) {
      // TODO add your handling code here:
      if(txtSearchByPrice.getText().equals("")){
                JOptionPane.showMessageDialog(rootPane,"Please insert price in the text
field to search.","Search Error",JOptionPane.INFORMATION_MESSAGE);
                return;
         }
         else if(list.isEmpty()){
                JOptionPane.showMessageDialog(rootPane,"There are no items found for searching!","Search Error",JOptionPane.INFORMATION_MESSAGE);

                return;

        }

        try

        {

                sort(list); //sorting list by sort method

                double searchByPrice = Double.parseDouble(txtSearchByPrice.getText().trim());

                int index = binarySearch(list,0,list.size()-1,searchByPrice);

                if(index==-1)

                {

                        JOptionPane.showMessageDialog(rootPane,"Accessory of price "+searchByPrice+" is not found.","Warning",JOptionPane.INFORMATION_MESSAGE);

                }

                else

                {

                        JOptionPane.showMessageDialog(rootPane,"The Model Number is "+list.get(index).getModelNumber()+",\n"

                        + "The accessory name is "+list.get(index).getAccessoryName()+",\n"

                        + "The Category is "+list.get(index).getCategory()+",\n"

                  + "The Recommendation is "+list.get(index).getRecommendation()+",\n"

                        + "The quality is "+list.get(index).getQuality()+",\n"
```

```
                                + "The Price is " +searchByPrice +" \n","Accessory
Found",JOptionPane.INFORMATION_MESSAGE);

                                //txtSearchByPrice.setText("");

                        }

                        toList();//calling toList method

                }

                catch(NumberFormatException nfe)

                {

                        JOptionPane.showMessageDialog(rootPane,"Please input price in number
in the text field.","Error",JOptionPane.ERROR_MESSAGE);

                }

    }


    private void txtSearchByPriceActionPerformed(java.awt.event.ActionEvent evt) {

        // TODO add your handling code here:

    }


    /**

     * @param args the command line arguments

     */

    public static void main(String args[]) {

        /* Set the Nimbus look and feel */
        //<editor-fold defaultstate="collapsed" desc=" Look and feel setting code (optional)
">

        /* If Nimbus (introduced in Java SE 6) is not available, stay with the default look and
feel.

         * For details see
http://download.oracle.com/javase/tutorial/uiswing/lookandfeel/plaf.html

         */

        try {

            for (javax.swing.UIManager.LookAndFeelInfo info :
javax.swing.UIManager.getInstalledLookAndFeels()) {

                if ("Nimbus".equals(info.getName())) {

                    javax.swing.UIManager.setLookAndFeel(info.getClassName());

                    javax.swing.UIManager.put("control",new java.awt.Color(204,255,204));

                    break;

                }

            }

        } catch (ClassNotFoundException ex) {


java.util.logging.Logger.getLogger(AppliancesInfo.class.getName()).log(java.util.logging.
Level.SEVERE, null, ex);

        } catch (InstantiationException ex) {


java.util.logging.Logger.getLogger(AppliancesInfo.class.getName()).log(java.util.logging.
Level.SEVERE, null, ex);

        } catch (IllegalAccessException ex) {
```

```
java.util.logging.Logger.getLogger(AppliancesInfo.class.getName()).log(java.util.logging.
Level.SEVERE, null, ex);
        } catch (javax.swing.UnsupportedLookAndFeelException ex) {

java.util.logging.Logger.getLogger(AppliancesInfo.class.getName()).log(java.util.logging.
Level.SEVERE, null, ex);
        }
        //</editor-fold>


        /* Create and display the form */
        java.awt.EventQueue.invokeLater(new Runnable() {
            public void run() {
                new AppliancesInfo().setVisible(true);
            }
        });
    }


    // Variables declaration - do not modify
    private javax.swing.JButton addBtn;
    private javax.swing.JButton btnDelete;
    private javax.swing.ButtonGroup buttonGroup1;
    private javax.swing.ButtonGroup buttonGroup2;
    private javax.swing.JPanel buttonPanel;
    private javax.swing.JButton clearBtn;
    private javax.swing.JComboBox<String> comboCategory;
    private javax.swing.JCheckBox commuCheckBox;
    private javax.swing.JCheckBox compaCheckBox;
    private javax.swing.JButton exitBtn;
    private javax.swing.JRadioButton highRadioBtn;
    private javax.swing.JPanel inputPanel;
    private javax.swing.JLabel jLabel1;
    private javax.swing.JLabel jLabel2;
    private javax.swing.JLabel jLabel3;
    private javax.swing.JLabel jLabel4;
    private javax.swing.JLabel jLabel6;
    private javax.swing.JLabel jLabel7;
    private javax.swing.JMenu jMenu1;
    private javax.swing.JMenu jMenu2;
    private javax.swing.JMenuBar jMenuBar1;
    private javax.swing.JMenuItem jMenuExit;
    private javax.swing.JMenuItem jMenuHelp;
    private javax.swing.JMenuItem jMenuOpen;
    private javax.swing.JScrollPane jScrollPane1;
    private javax.swing.JPopupMenu.Separator jSeparator1;
    private javax.swing.JRadioButton lowRadioBtn;
    private javax.swing.JRadioButton medRadioBtn;
    private javax.swing.JCheckBox norecoCheckBox;

    private javax.swing.JButton searchBtn;

    private javax.swing.JButton searchCategoryBtn;

    private javax.swing.JComboBox<String> searchCategoryCmbBox;

    private javax.swing.JPanel searchPanel;

    private javax.swing.JPanel tablePanel;

    private javax.swing.JTable tblDetails;

    private javax.swing.JTextField txtAccName;

    private javax.swing.JTextField txtModelNum;

    private javax.swing.JTextField txtPrice;

    private javax.swing.JTextField txtSearchByPrice;

    // End of variables declaration


    private void clear() {

        throw new UnsupportedOperationException("Not supported yet."); //To change body
of generated methods, choose Tools | Templates.

    }

}/*

 * To change this license header, choose License Headers in Project Properties.

 * To change this template file, choose Tools | Templates

 * and open the template in the editor.

 */

package AppliancesInfo;
```

```
/**
 *
 * @author Delish Khadka
 */
public class PhoneDetails {
    private String modelNumber,accessoryName,category,recommendation,quality;
    private double price;

    public PhoneDetails(String modelNumber,String accessoryName, String category, String recommendation, String quality, double price) {
        this.modelNumber = modelNumber;
        this.accessoryName = accessoryName;
        this.category = category;
        this.recommendation = recommendation;
        this.quality = quality;
        this.price = price;

}

    public String getModelNumber() {
        return modelNumber;
    }
    public String getAccessoryName() {
        return accessoryName;
    }

    public String getCategory() {
        return category;
    }

     public String getRecommendation() {
        return recommendation;
    }

    public String getQuality() {
        return quality;
    }

    public double getPrice() {
        return price;
    }

}
```